Lab 9 - NIDS/NIPS and Web Proxy Analysis

University of Maryland Baltimore County

Presented to: (professor's name removed)

April 17, 2020

Table of Content

Introduction

While working with Division of IT at University of Maryland Baltimore County, for their Digital Forensics and Incident Response team, I have been tasked to a) Perform log analysis on IDS/IPS alerts (Snort) to become familiar with alert formats and using them to investigate potentially malicious traffic and b) Perform log analysis on Proxy logs (Squid) to become familiar with proxy log formats, and how to review them to investigate web related traffic. The methodology adopted for the analysis of the given alert, tcpdump.log, snort.conf, access.log and store.log files is done by using a combination of tools and linux terminal commands– wireshark, bless hex editor, grep command and MS Excel. This task encompasses in detail analysis of the logs and alerts generated by Network Intrusion Protection and Detection Systems (NIDS/NIPS).

In this fictitious scenario, you'll be provided IDS/IPS alert data and proxy logs in an effort to assist an investigation.

In his quest to save the planet, Inter0ptic has started a credit card number recycling program. "Do you have a database filled with credit card numbers, just sitting there collecting dust? Put that data to good use!" he writes on his website. "Recycle your company's used credit card numbers! Send us your database, and we'll send YOU a check." For good measure, .Inter0ptic decides to add some bells and whistles to the site too.

Meanwhile…. MacDaddy Payment Processor has deployed Snort NIDS sensors to detect an array of anomalous events, bot inbound and outbound. An alert was logged at 08:01:45 on 5/18/11 concerning an inbound chunk of executable code sent to port 80/tcp for inside host 192.168.1.169 from external host 172.16.16.218.

Alert:

```
[] [1:10000648:2] SHELLCODE x86 NOOP [] [Classification: Executable code was detected][Priority: 1] 5/18-08:01:45.591840 172.16.16.218:80 -> 192.168.1.169:2493 TCP TTL: 63 TOS:0x0 ID:53309 IpLen: 20 DgmLen: 1127 DF AP Seq: 0x1B2C3517 Ack: 0x9F9E0666 Win: 0x1920 TcpLen: 20
```

Network:

Internal network: 192.168.1.0/24 DMZ network: 10.1.1.0/24 "Internet": 172.16.16.0/24 Other domains and subnets of interest include: .evl – a top level domain used by Evil Systems Example.com – MacDaddy Payment Processor local domain

The security staff at MacDaddy Payment Processor collects the Snort alerts for the day in question and preserves the "tcpdump.log" file that corresponds with the alerts. At your request, they also gather the relevant Snort sensor's config and rules. You are provided with the following files containing data to analyze:

- **Alert** – A text file containing the Snort sensor's default "alert" output, including the alert of interest above
- **Tcpdump.log** – A libpcap generated file that contains full-content packet captures of the packets involved in the events summarized in the above alert file.
- **Snort.conf** – A text file containing the configuration description of the Snort sensor, including the rules
- **Rules** – A folder containing the Snort rules that were in use by the sensor, as included by the configuration above.

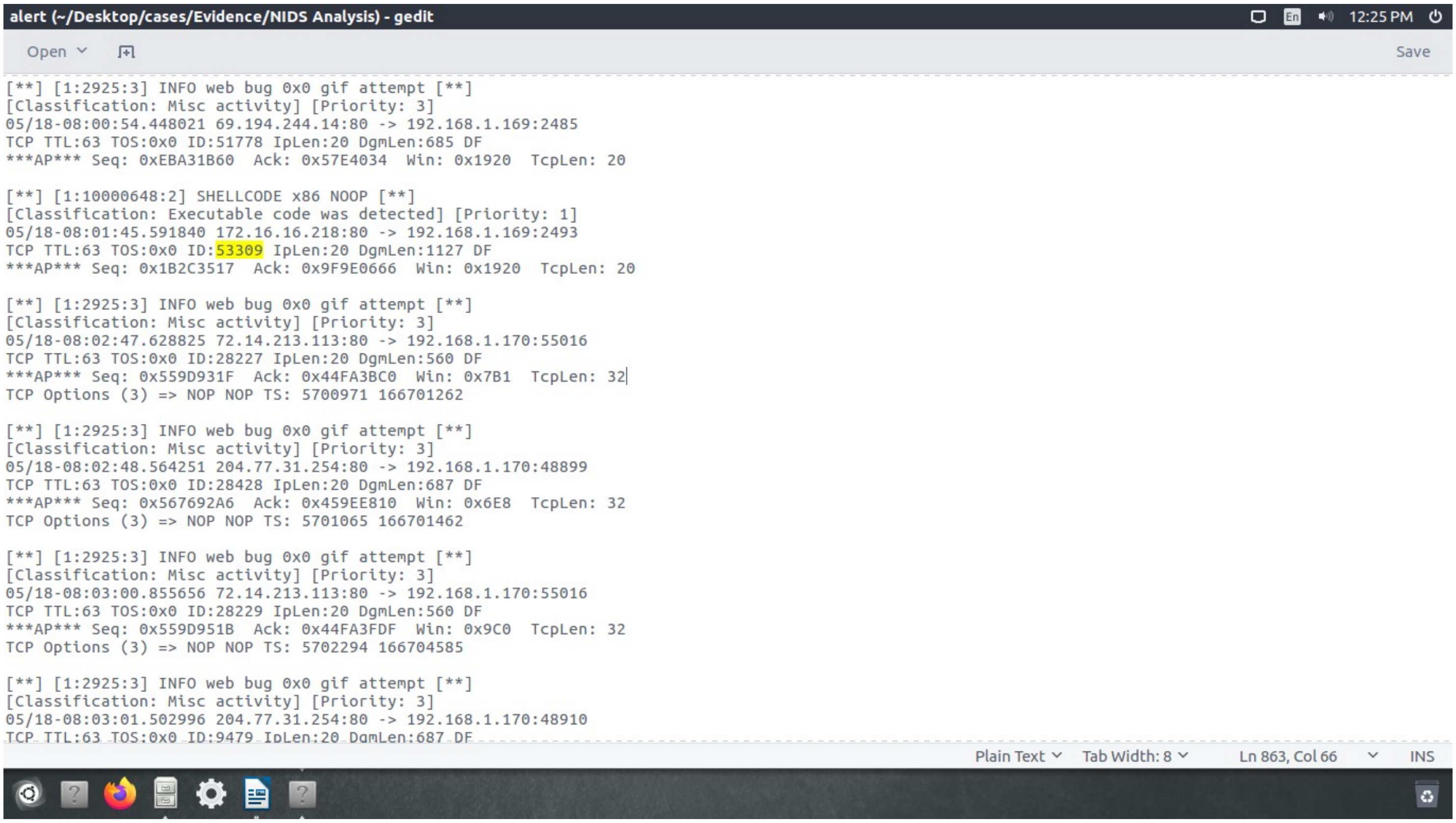
All files are located under the /cases/Evidence/ within the IDS/IPS Alert Analysis folder.

Analysis

1. IDS/IPS Alert Background (Part 1)

1.1 Examine the alert's data to understand the logistical context

Figure 1.1.1, we analyse the alert and note: Source IP: 172.16.16.218 (Source Port :80) and Destination IP: 192.168.1.169:2493, Packet Delivery ID : 53309, Snort ID: 10000648, IPlink : 20 and Datagram link : 1127

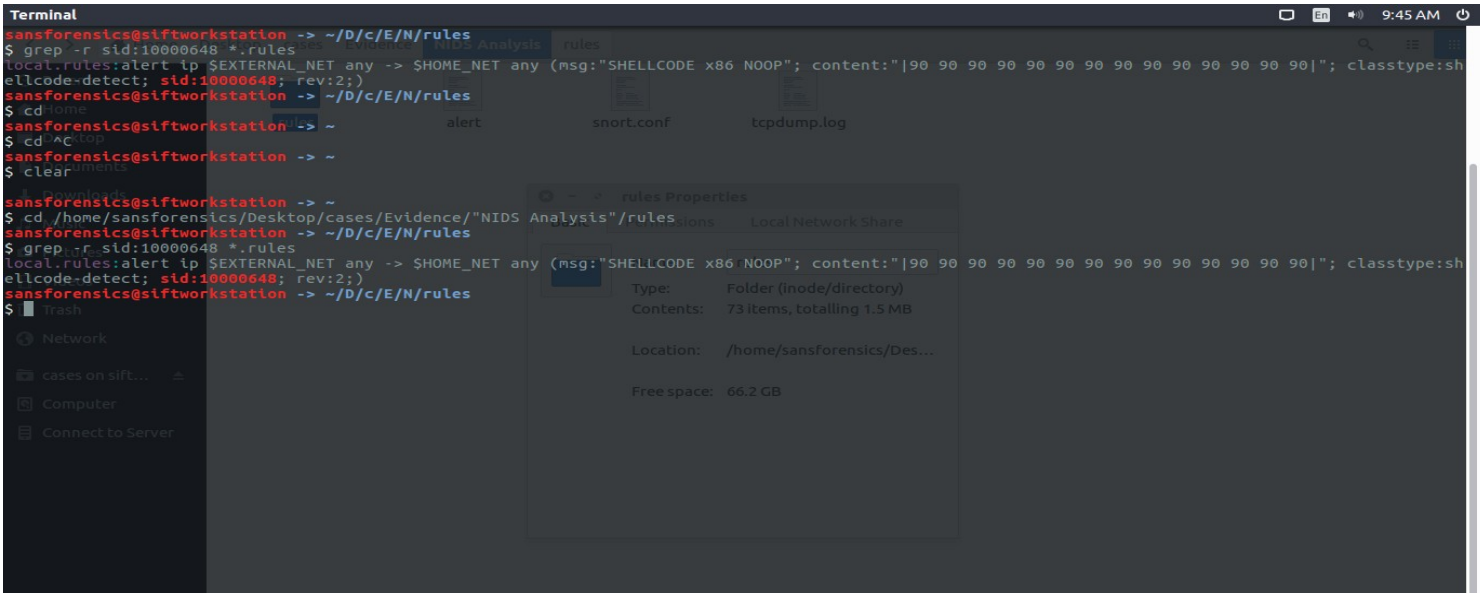


```
[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:00:54.448021 69.194.244.14:80 -> 192.168.1.169:2485
TCP TTL:63 TOS:0x0 ID:51778 IpLen:20 DgmLen:685 DF
***AP*** Seq: 0xEBA31B60  Ack: 0x57E4034  Win: 0x1920  TcpLen: 20

[**] [1:10000648:2] SHELLCODE x86 NOOP [**]
[Classification: Executable code was detected] [Priority: 1]
05/18-08:01:45.591840 172.16.16.218:80 -> 192.168.1.169:2493
TCP TTL:63 TOS:0x0 ID:53309 IpLen:20 DgmLen:1127 DF
***AP*** Seq: 0x1B2C3517  Ack: 0x9F9E0666  Win: 0x1920  TcpLen: 20

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:02:47.628825 72.14.213.113:80 -> 192.168.1.170:55016
TCP TTL:63 TOS:0x0 ID:28227 IpLen:20 DgmLen:560 DF
***AP*** Seq: 0x559D931F  Ack: 0x44FA3BC0  Win: 0x7B1  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5700971 166701262

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:02:48.564251 204.77.31.254:80 -> 192.168.1.170:48899
TCP TTL:63 TOS:0x0 ID:28428 IpLen:20 DgmLen:687 DF
***AP*** Seq: 0x567692A6  Ack: 0x459EE810  Win: 0x6E8  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5701065 166701462

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:03:00.855656 72.14.213.113:80 -> 192.168.1.170:55016
TCP TTL:63 TOS:0x0 ID:28229 IpLen:20 DgmLen:560 DF
***AP*** Seq: 0x559D951B  Ack: 0x44FA3FDF  Win: 0x9C0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5702294 166704585

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:03:01.502996 204.77.31.254:80 -> 192.168.1.170:48910
TCP TTL:63 TOS:0x0 ID:9479 IpLen:20 DgmLen:687 DF
```

Figure 1.1.1

1.2 Compare the alert to the rule, in order to better understand WHAT it has been built to detect:

Figure 1.2.1, suing the snort ID we use **cd /home/sansforensics/Desktop/cases/Evidence/"NIDS Analysis"/rules** and then **grep -r sid:10000648 *.rules** to find the **shellcode-**detect among a list of .rules files and understand more about alert and the rule using the sid and find that **local.rule** file mentions the **classtype : shellcode-detect**

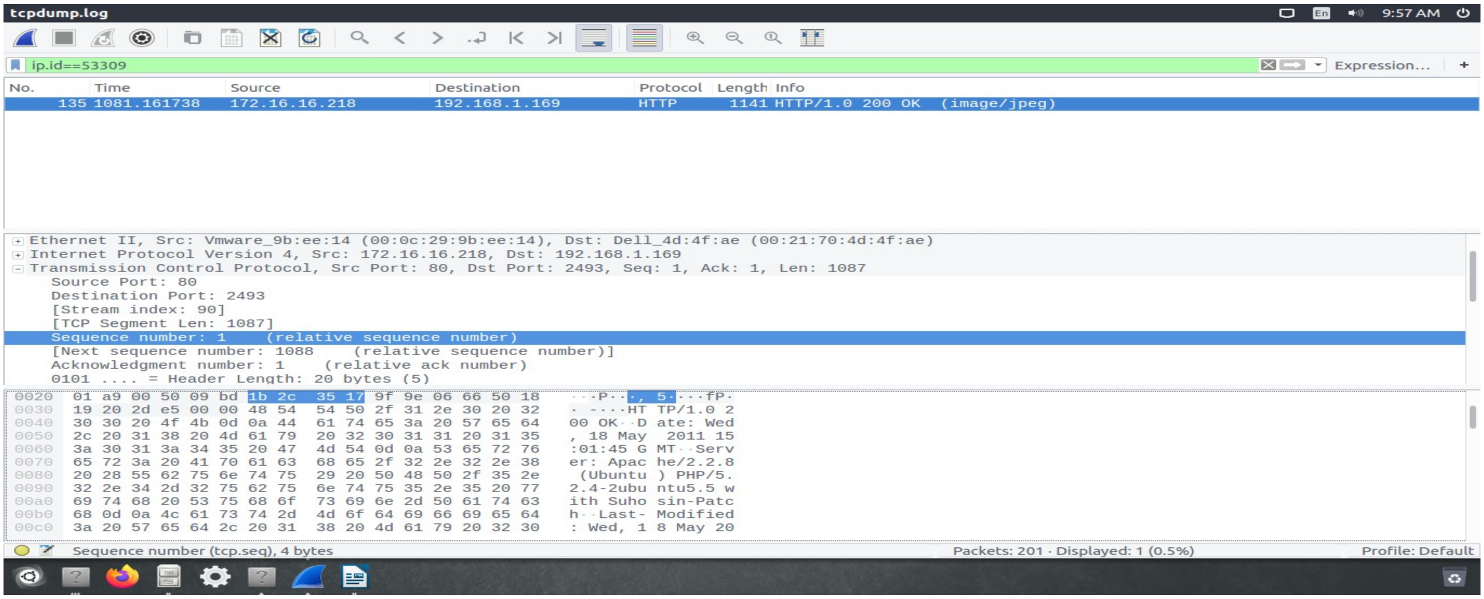


```
sansforensics@siftworkstation -> ~/D/c/E/N/rules
$ grep -r sid:10000648 *.rules
local.rules:alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"SHELLCODE x86 NOOP"; content:"|90 90 90 90 90 90 90 90 90 90 90 90 90 90|"; classtype:sh
ellcode-detect; sid:10000648; rev:2;)
sansforensics@siftworkstation -> ~/D/c/E/N/rules
$ cd
sansforensics@siftworkstation -> ~
$ cd ^C
sansforensics@siftworkstation -> ~
$ clear

sansforensics@siftworkstation -> ~
$ cd /home/sansforensics/Desktop/cases/Evidence/"NIDS Analysis"/rules
sansforensics@siftworkstation -> ~/D/c/E/N/rules
$ grep -r sid:10000648 *.rules
local.rules:alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"SHELLCODE x86 NOOP"; content:"|90 90 90 90 90 90 90 90 90 90 90 90 90 90|"; classtype:sh
ellcode-detect; sid:10000648; rev:2;)
sansforensics@siftworkstation -> ~/D/c/E/N/rules
$
```

Figure 1.2.1

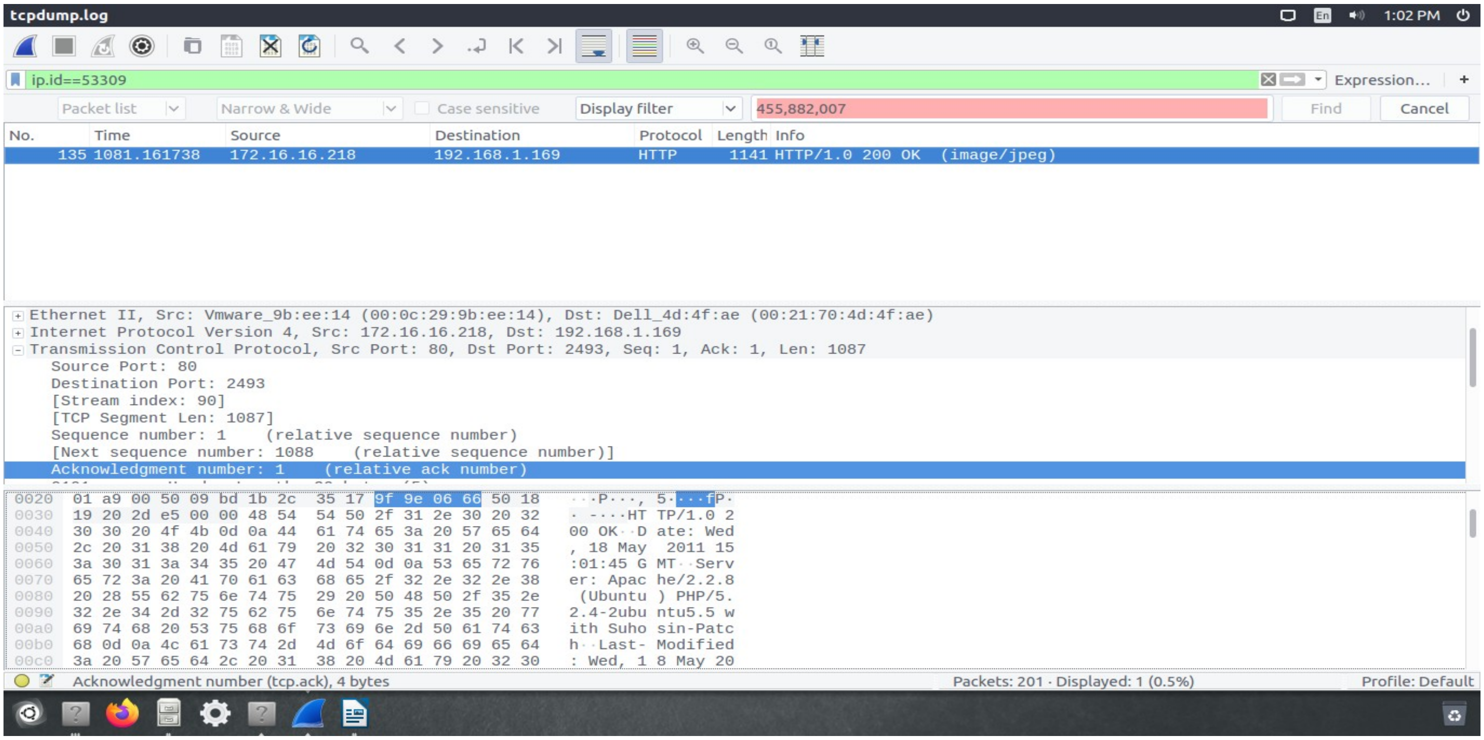
1.3 Retrieve the packet that triggered the alert

Figure 1.3.1 and 1.3.2, we open the **tcpdump.log** file in wireshark and use the filter **ip.id=-53309** to analyse the packet and we were able to find **Seq: 0x1B2C3517** and **Ack: 0x9F9E0666** within TCP hex dump.



```
ip.id==53309

No.   Time          Source          Destination     Protocol Length Info
  135 1081.161738   172.16.16.218   192.168.1.169   HTTP       1141 HTTP/1.0 200 OK  (image/jpeg)

+ Ethernet II, Src: Vmware_9b:ee:14 (00:0c:29:9b:ee:14), Dst: Dell_4d:4f:ae (00:21:70:4d:4f:ae)
+ Internet Protocol Version 4, Src: 172.16.16.218, Dst: 192.168.1.169
- Transmission Control Protocol, Src Port: 80, Dst Port: 2493, Seq: 1, Ack: 1, Len: 1087
     Source Port: 80
     Destination Port: 2493
     [Stream index: 90]
     [TCP Segment Len: 1087]
     Sequence number: 1     (relative sequence number)
     [Next sequence number: 1088     (relative sequence number)]
     Acknowledgment number: 1     (relative ack number)
     0101 .... = Header Length: 20 bytes (5)

0020   01 a9 00 50 09 bd 1b 2c  35 17 9f 9e 06 66 50 18   ···P··,  5····fP·
0030   19 20 2d e5 00 00 48 54  54 50 2f 31 2e 30 20 32   · -···HT TP/1.0 2
0040   30 30 20 4f 4b 0d 0a 44  61 74 65 3a 20 57 65 64   00 OK··D ate: Wed
0050   2c 20 31 38 20 4d 61 79  20 32 30 31 31 20 31 35   , 18 May  2011 15
0060   3a 30 31 3a 34 35 20 47  4d 54 0d 0a 53 65 72 76   :01:45 G MT··Serv
0070   65 72 3a 20 41 70 61 63  68 65 2f 32 2e 32 2e 38   er: Apac he/2.2.8
0080   20 28 55 62 75 6e 74 75  29 20 50 48 50 2f 35 2e    (Ubuntu ) PHP/5.
0090   32 2e 34 2d 32 75 62 75  6e 74 75 35 2e 35 20 77   2.4-2ubu ntu5.5 w
00a0   69 74 68 20 53 75 68 6f  73 69 6e 2d 50 61 74 63   ith Suho sin-Patc
00b0   68 0d 0a 4c 61 73 74 2d  4d 6f 64 69 66 69 65 64   h··Last- Modified
00c0   3a 20 57 65 64 2c 20 31  38 20 4d 61 79 20 32 30   : Wed, 1 8 May 20

Sequence number (tcp.seq), 4 bytes          Packets: 201 · Displayed: 1 (0.5%)          Profile: Default
```

Figure 1.3.1



```
No.   Time          Source         Destination     Protocol  Length Info
  135 1081.161738   172.16.16.218  192.168.1.169   HTTP        1141 HTTP/1.0 200 OK  (image/jpeg)
```

```
+ Ethernet II, Src: Vmware_9b:ee:14 (00:0c:29:9b:ee:14), Dst: Dell_4d:4f:ae (00:21:70:4d:4f:ae)
+ Internet Protocol Version 4, Src: 172.16.16.218, Dst: 192.168.1.169
- Transmission Control Protocol, Src Port: 80, Dst Port: 2493, Seq: 1, Ack: 1, Len: 1087
    Source Port: 80
    Destination Port: 2493
    [Stream index: 90]
    [TCP Segment Len: 1087]
    Sequence number: 1    (relative sequence number)
    [Next sequence number: 1088    (relative sequence number)]
    Acknowledgment number: 1    (relative ack number)
```

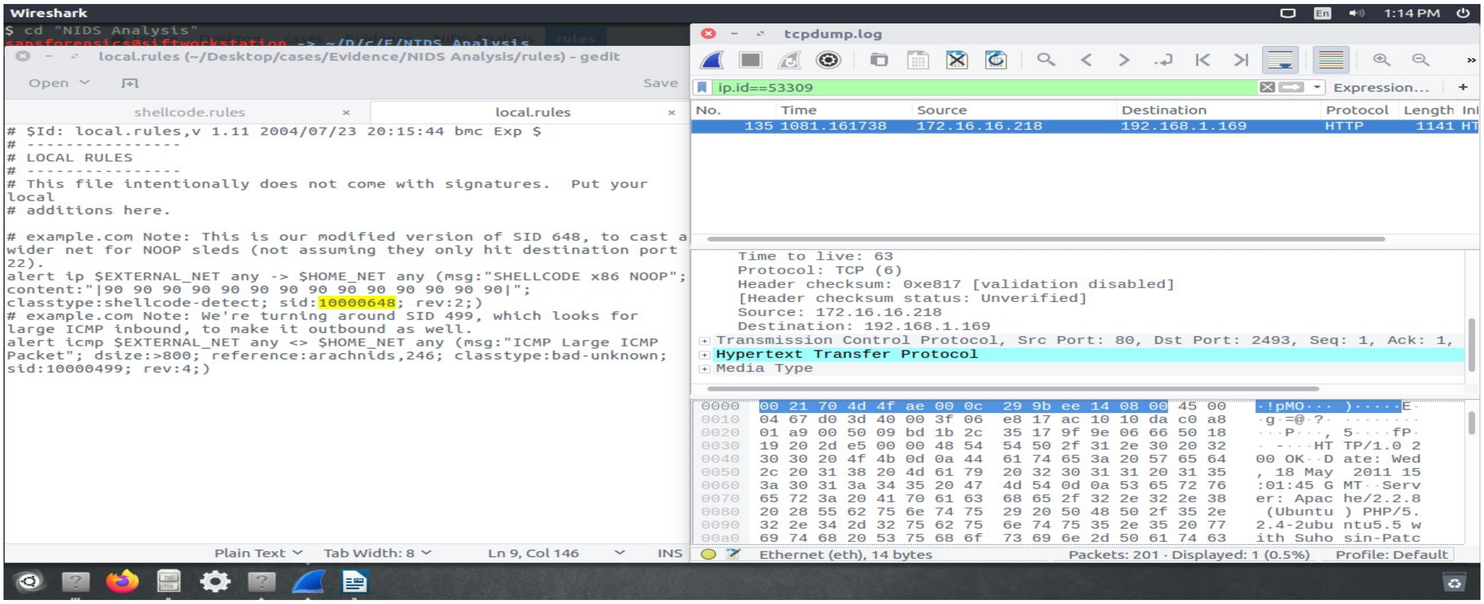
```
0020  01 a9 00 50 09 bd 1b 2c  35 17 9f 9e 06 66 50 18   ···P···, 5····fP·
0030  19 20 2d e5 00 00 48 54  54 50 2f 31 2e 30 20 32   · -···HT TP/1.0 2
0040  30 30 20 4f 4b 0d 0a 44  61 74 65 3a 20 57 65 64   00 OK··D ate: Wed
0050  2c 20 31 38 20 4d 61 79  20 32 30 31 31 20 31 35   , 18 May  2011 15
0060  3a 30 31 3a 34 35 20 47  4d 54 0d 0a 53 65 72 76   :01:45 G MT··Serv
0070  65 72 3a 20 41 70 61 63  68 65 2f 32 2e 32 2e 38   er: Apac he/2.2.8
0080  20 28 55 62 75 6e 74 75  29 20 50 48 50 2f 35 2e    (Ubuntu ) PHP/5.
0090  32 2e 34 2d 32 75 62 75  6e 74 75 35 2e 35 20 77   2.4-2ubu ntu5.5 w
00a0  69 74 68 20 53 75 68 6f  73 69 6e 2d 50 61 74 63   ith Suho sin-Patc
00b0  68 0d 0a 4c 61 73 74 2d  4d 6f 64 69 66 69 65 64   h··Last- Modified
00c0  3a 20 57 65 64 2c 20 31  38 20 4d 61 79 20 32 30   : Wed, 1 8 May 20
```

Acknowledgment number (tcp.ack), 4 bytes | Packets: 201 · Displayed: 1 (0.5%) | Profile: Default

Figure 1.3.2

1.4 Compare the rule to the packet to understand WHY it fired

Figure 1.4.1 and 1.4.2, we compare the rule and packet and find that the rule is set to detect suspicious content from an external IP to port 80 and here an image file is being transferred with potentially malicious shellcode.



```
$ cd "NIDS Analysis"
sansforensics@siftworkstation -> ~/D/c/E/NIDS Analysis
```

```
# $Id: local.rules,v 1.11 2004/07/23 20:15:44 bmc Exp $
# ----------------
# LOCAL RULES
# ----------------
# This file intentionally does not come with signatures.  Put your
local
# additions here.

# example.com Note: This is our modified version of SID 648, to cast a
wider net for NOOP sleds (not assuming they only hit destination port
22).
alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"SHELLCODE x86 NOOP";
content:"|90 90 90 90 90 90 90 90 90 90 90 90 90 90|";
classtype:shellcode-detect; sid:10000648; rev:2;)
# example.com Note: We're turning around SID 499, which looks for
large ICMP inbound, to make it outbound as well.
alert icmp $EXTERNAL_NET any <> $HOME_NET any (msg:"ICMP Large ICMP
Packet"; dsize:>800; reference:arachnids,246; classtype:bad-unknown;
sid:10000499; rev:4;)
```

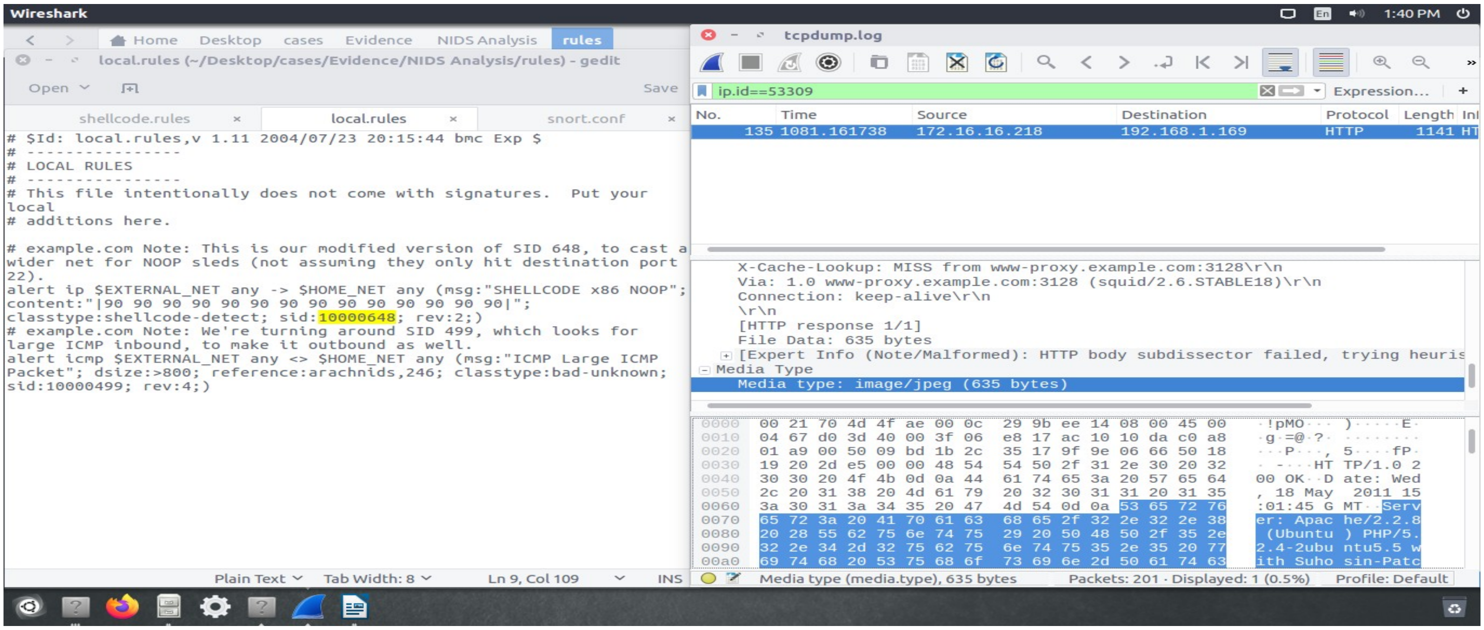
```
No.   Time          Source         Destination     Protocol  Length In
  135 1081.161738   172.16.16.218  192.168.1.169   HTTP        1141 HT
```

```
    Time to live: 63
    Protocol: TCP (6)
    Header checksum: 0xe817 [validation disabled]
    [Header checksum status: Unverified]
    Source: 172.16.16.218
    Destination: 192.168.1.169
+ Transmission Control Protocol, Src Port: 80, Dst Port: 2493, Seq: 1, Ack: 1,
+ Hypertext Transfer Protocol
+ Media Type
```

```
0000  00 21 70 4d 4f ae 00 0c  29 9b ee 14 08 00 45 00   ·!pMO··· )·····E·
0010  04 67 d0 3d 40 00 3f 06  e8 17 ac 10 10 da c0 a8   ·g·=@·?· ········
0020  01 a9 00 50 09 bd 1b 2c  35 17 9f 9e 06 66 50 18   ···P···, 5····fP·
0030  19 20 2d e5 00 00 48 54  54 50 2f 31 2e 30 20 32   · -···HT TP/1.0 2
0040  30 30 20 4f 4b 0d 0a 44  61 74 65 3a 20 57 65 64   00 OK··D ate: Wed
0050  2c 20 31 38 20 4d 61 79  20 32 30 31 31 20 31 35   , 18 May  2011 15
0060  3a 30 31 3a 34 35 20 47  4d 54 0d 0a 53 65 72 76   :01:45 G MT··Serv
0070  65 72 3a 20 41 70 61 63  68 65 2f 32 2e 32 2e 38   er: Apac he/2.2.8
0080  20 28 55 62 75 6e 74 75  29 20 50 48 50 2f 35 2e    (Ubuntu ) PHP/5.
0090  32 2e 34 2d 32 75 62 75  6e 74 75 35 2e 35 20 77   2.4-2ubu ntu5.5 w
00a0  69 74 68 20 53 75 68 6f  73 69 6e 2d 50 61 74 63   ith Suho sin-Patc
```

Ethernet (eth), 14 bytes | Packets: 201 · Displayed: 1 (0.5%) | Profile: Default

Figure 1.4.1



```
# $Id: local.rules,v 1.11 2004/07/23 20:15:44 bmc Exp $
# ----------------
# LOCAL RULES
# ----------------
# This file intentionally does not come with signatures.  Put your
local
# additions here.

# example.com Note: This is our modified version of SID 648, to cast a
wider net for NOOP sleds (not assuming they only hit destination port
22).
alert ip $EXTERNAL_NET any -> $HOME_NET any (msg:"SHELLCODE x86 NOOP";
content:"|90 90 90 90 90 90 90 90 90 90 90 90 90 90|";
classtype:shellcode-detect; sid:10000648; rev:2;)
# example.com Note: We're turning around SID 499, which looks for
large ICMP inbound, to make it outbound as well.
alert icmp $EXTERNAL_NET any <> $HOME_NET any (msg:"ICMP Large ICMP
Packet"; dsize:>800; reference:arachnids,246; classtype:bad-unknown;
sid:10000499; rev:4;)
```

```
ip.id==53309
```

No.	Time	Source	Destination	Protocol	Length	Inf
135	1081.161738	172.16.16.218	192.168.1.169	HTTP	1141	HT

```
    X-Cache-Lookup: MISS from www-proxy.example.com:3128\r\n
    Via: 1.0 www-proxy.example.com:3128 (squid/2.6.STABLE18)\r\n
    Connection: keep-alive\r\n
    \r\n
    [HTTP response 1/1]
    File Data: 635 bytes
  + [Expert Info (Note/Malformed): HTTP body subdissector failed, trying heuris
- Media Type
    Media type: image/jpeg (635 bytes)
```

```
0000  00 21 70 4d 4f ae 00 0c  29 9b ee 14 08 00 45 00   ·!pMO···  )·····E·
0010  04 67 d0 3d 40 00 3f 06  e8 17 ac 10 10 da c0 a8   ·g·=@·?·  ········
0020  01 a9 00 50 09 bd 1b 2c  35 17 9f 9e 06 66 50 18   ···P···,  5····fP·
0030  19 20 2d e5 00 00 48 54  54 50 2f 31 2e 30 20 32   · -···HT  TP/1.0 2
0040  30 30 20 4f 4b 0d 0a 44  61 74 65 3a 20 57 65 64   00 OK··D  ate: Wed
0050  2c 20 31 38 20 4d 61 79  20 32 30 31 31 20 31 35   , 18 May   2011 15
0060  3a 30 31 3a 34 35 20 47  4d 54 0d 0a 53 65 72 76   :01:45 G  MT··Serv
0070  65 72 3a 20 41 70 61 63  68 65 2f 32 2e 32 2e 38   er: Apac  he/2.2.8
0080  20 28 55 62 75 6e 74 75  29 20 50 48 50 2f 35 2e    (Ubuntu  ) PHP/5.
0090  32 2e 34 2d 32 75 62 75  6e 74 75 35 2e 35 20 77   2.4-2ubu  ntu5.5 w
00a0  69 74 68 20 53 75 68 6f  73 69 6e 2d 50 61 74 63   ith Suho  sin-Patc
```

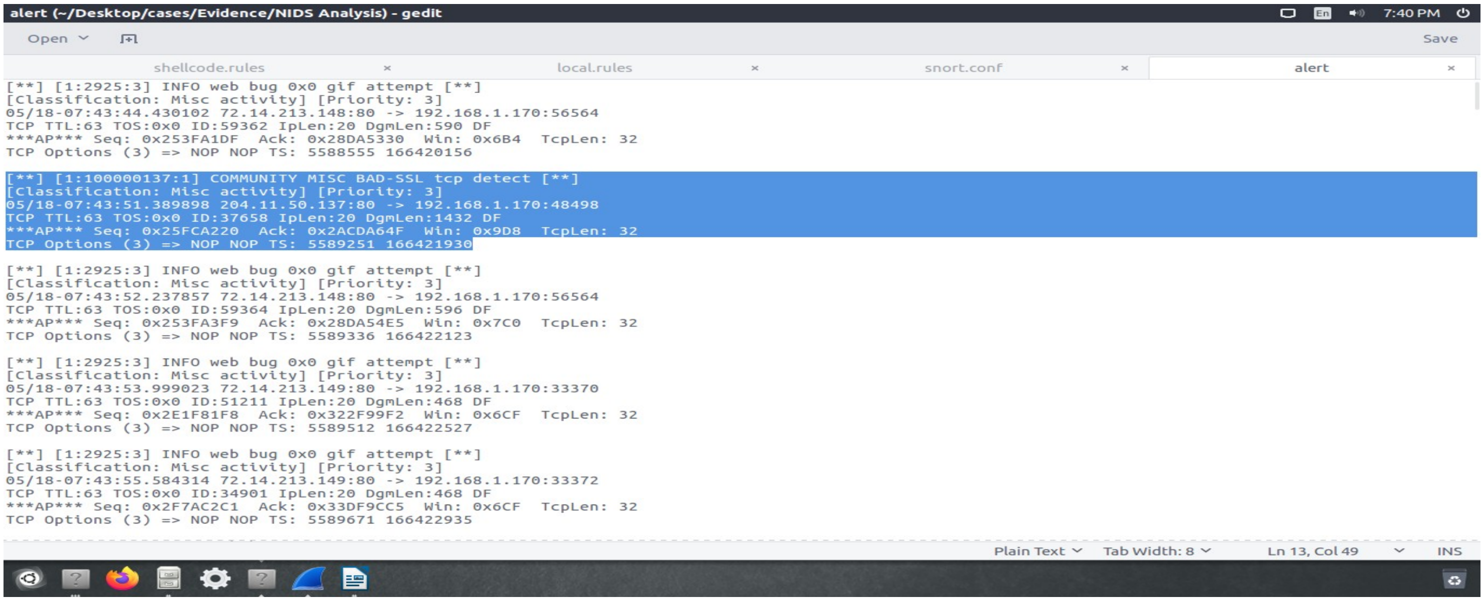
```
Media type (media.type), 635 bytes    Packets: 201 · Displayed: 1 (0.5%)    Profile: Default
```

Figure 1.4.2

1.5 Construct a timeline of alerted activities involving the potentially malicious outside host

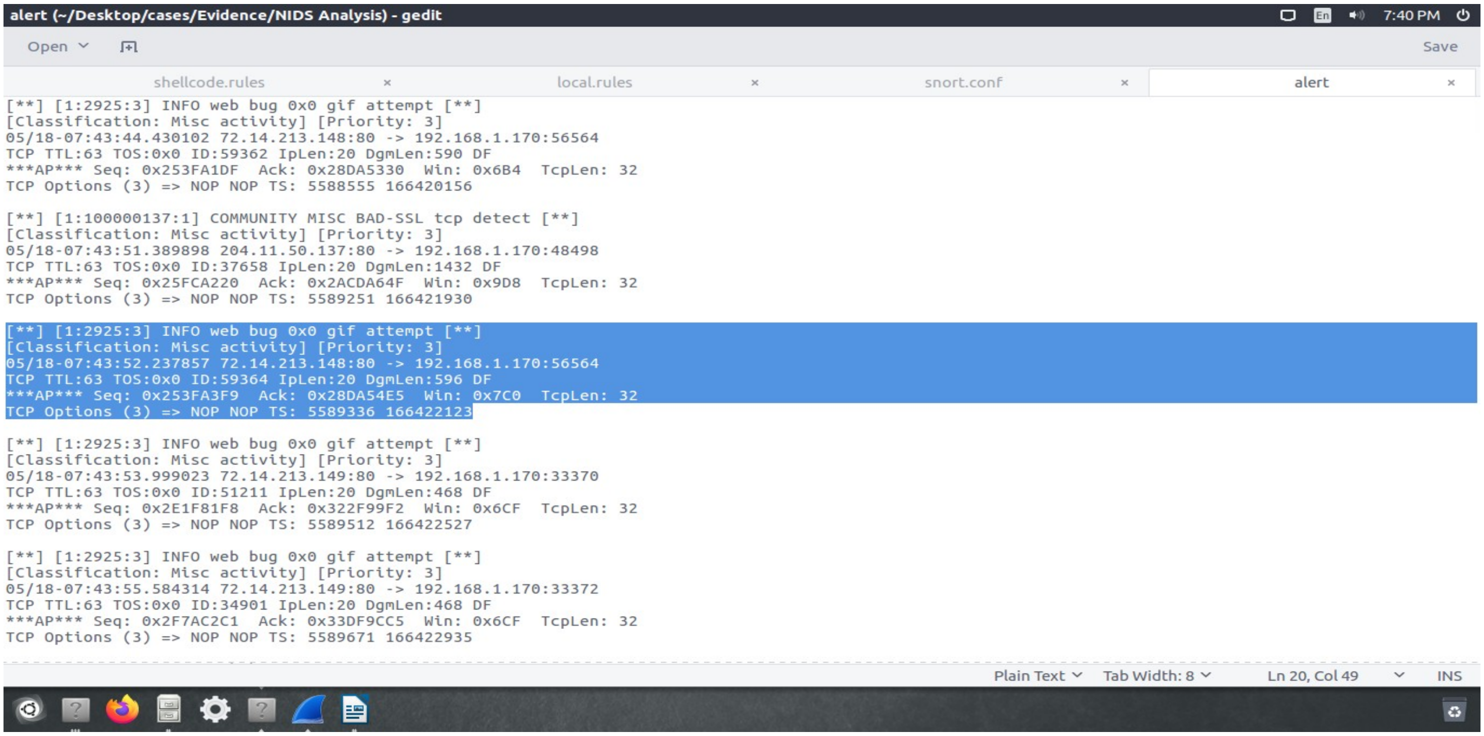
In Figure 1.5.1, 1.5.2, 1.5.3 and 1.5.4:

- at 07:43:51 we detect "COMMUNITY MISC BAD-SSL tcp detect [**] [Classification: Misc activity]" from 204.11.50.137:80 to internal host 192.168.1.169
- From 07:43:52 to 08:15:08, we detect "INFO web bug 0x0 gif attempt [**] [Classification: Misc activity]" from different external hosts to internal hosts like 192.168.1.169, 192.168.1.170etc
- In between the above time frame, at 08:01:45 MST we also detect "SHELLCODE x86 NOOP [**] [Classification: Executable code was detected]", from an external web server 172.16.16.218:80 to 192.168.1.169



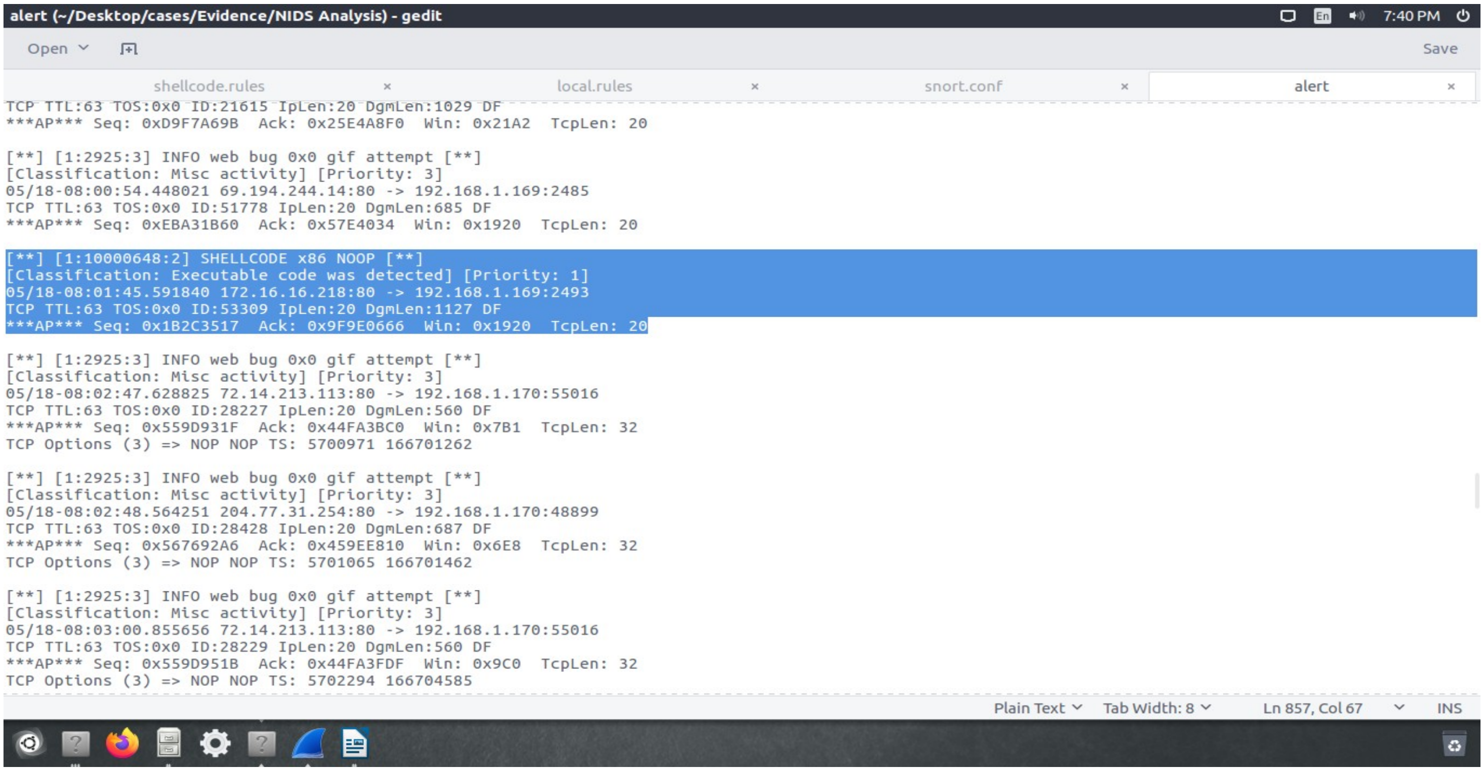
```
[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:43:44.430102 72.14.213.148:80 -> 192.168.1.170:56564
TCP TTL:63 TOS:0x0 ID:59362 IpLen:20 DgmLen:590 DF
***AP*** Seq: 0x253FA1DF  Ack: 0x28DA5330  Win: 0x6B4  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5588555 166420156

[**] [1:100000137:1] COMMUNITY MISC BAD-SSL tcp detect [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:43:51.389898 204.11.50.137:80 -> 192.168.1.170:48498
TCP TTL:63 TOS:0x0 ID:37658 IpLen:20 DgmLen:1432 DF
***AP*** Seq: 0x25FCA220  Ack: 0x2ACDA64F  Win: 0x9D8  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5589251 166421930

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:43:52.237857 72.14.213.148:80 -> 192.168.1.170:56564
TCP TTL:63 TOS:0x0 ID:59364 IpLen:20 DgmLen:596 DF
***AP*** Seq: 0x253FA3F9  Ack: 0x28DA54E5  Win: 0x7C0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5589336 166422123

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:43:53.999023 72.14.213.149:80 -> 192.168.1.170:33370
TCP TTL:63 TOS:0x0 ID:51211 IpLen:20 DgmLen:468 DF
***AP*** Seq: 0x2E1F81F8  Ack: 0x322F99F2  Win: 0x6CF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5589512 166422527

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:43:55.584314 72.14.213.149:80 -> 192.168.1.170:33372
TCP TTL:63 TOS:0x0 ID:34901 IpLen:20 DgmLen:468 DF
***AP*** Seq: 0x2F7AC2C1  Ack: 0x33DF9CC5  Win: 0x6CF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5589671 166422935
```

Figure 1.5.1



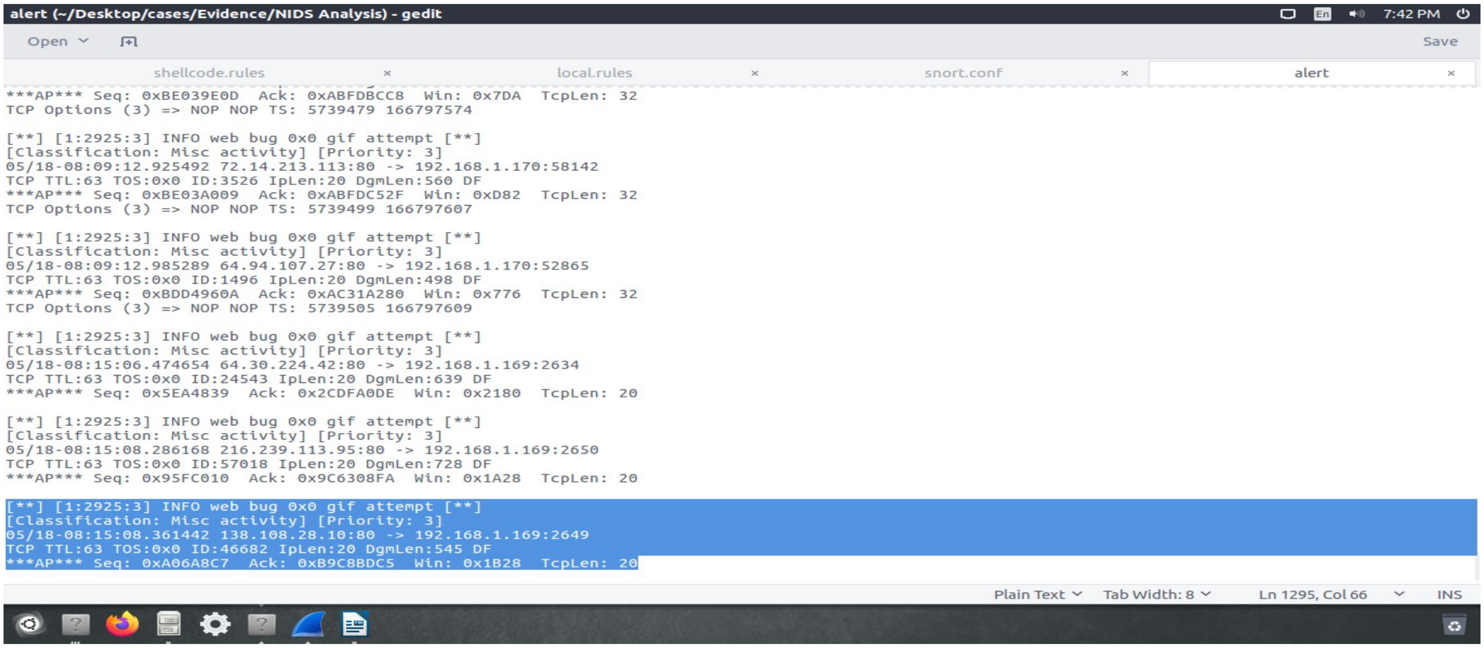
```
[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:43:44.430102 72.14.213.148:80 -> 192.168.1.170:56564
TCP TTL:63 TOS:0x0 ID:59362 IpLen:20 DgmLen:590 DF
***AP*** Seq: 0x253FA1DF  Ack: 0x28DA5330  Win: 0x6B4  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5588555 166420156

[**] [1:100000137:1] COMMUNITY MISC BAD-SSL tcp detect [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:43:51.389898 204.11.50.137:80 -> 192.168.1.170:48498
TCP TTL:63 TOS:0x0 ID:37658 IpLen:20 DgmLen:1432 DF
***AP*** Seq: 0x25FCA220  Ack: 0x2ACDA64F  Win: 0x9D8  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5589251 166421930

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:43:52.237857 72.14.213.148:80 -> 192.168.1.170:56564
TCP TTL:63 TOS:0x0 ID:59364 IpLen:20 DgmLen:596 DF
***AP*** Seq: 0x253FA3F9  Ack: 0x28DA54E5  Win: 0x7C0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5589336 166422123

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:43:53.999023 72.14.213.149:80 -> 192.168.1.170:33370
TCP TTL:63 TOS:0x0 ID:51211 IpLen:20 DgmLen:468 DF
***AP*** Seq: 0x2E1F81F8  Ack: 0x322F99F2  Win: 0x6CF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5589512 166422527

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:43:55.584314 72.14.213.149:80 -> 192.168.1.170:33372
TCP TTL:63 TOS:0x0 ID:34901 IpLen:20 DgmLen:468 DF
***AP*** Seq: 0x2F7AC2C1  Ack: 0x33DF9CC5  Win: 0x6CF  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5589671 166422935
```

Figure 1.5.2



```
TCP TTL:63 TOS:0x0 ID:21615 IpLen:20 DgmLen:1029 DF
***AP*** Seq: 0xD9F7A69B  Ack: 0x25E4A8F0  Win: 0x21A2  TcpLen: 20

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:00:54.448021 69.194.244.14:80 -> 192.168.1.169:2485
TCP TTL:63 TOS:0x0 ID:51778 IpLen:20 DgmLen:685 DF
***AP*** Seq: 0xEBA31B60  Ack: 0x57E4034  Win: 0x1920  TcpLen: 20

[**] [1:10000648:2] SHELLCODE x86 NOOP [**]
[Classification: Executable code was detected] [Priority: 1]
05/18-08:01:45.591840 172.16.16.218:80 -> 192.168.1.169:2493
TCP TTL:63 TOS:0x0 ID:53309 IpLen:20 DgmLen:1127 DF
***AP*** Seq: 0x1B2C3517  Ack: 0x9F9E0666  Win: 0x1920  TcpLen: 20

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:02:47.628825 72.14.213.113:80 -> 192.168.1.170:55016
TCP TTL:63 TOS:0x0 ID:28227 IpLen:20 DgmLen:560 DF
***AP*** Seq: 0x559D931F  Ack: 0x44FA3BC0  Win: 0x7B1  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5700971 166701262

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:02:48.564251 204.77.31.254:80 -> 192.168.1.170:48899
TCP TTL:63 TOS:0x0 ID:28428 IpLen:20 DgmLen:687 DF
***AP*** Seq: 0x567692A6  Ack: 0x459EE810  Win: 0x6E8  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5701065 166701462

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:03:00.855656 72.14.213.113:80 -> 192.168.1.170:55016
TCP TTL:63 TOS:0x0 ID:28229 IpLen:20 DgmLen:560 DF
***AP*** Seq: 0x559D951B  Ack: 0x44FA3FDF  Win: 0x9C0  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5702294 166704585
```

Figure 1.5.3



```
***AP*** Seq: 0xBE039E0D  Ack: 0xABFDBCC8  Win: 0x7DA  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5739479 166797574

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:09:12.925492 72.14.213.113:80 -> 192.168.1.170:58142
TCP TTL:63 TOS:0x0 ID:3526 IpLen:20 DgmLen:560 DF
***AP*** Seq: 0xBE03A009  Ack: 0xABFDC52F  Win: 0xD82  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5739499 166797607

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:09:12.985289 64.94.107.27:80 -> 192.168.1.170:52865
TCP TTL:63 TOS:0x0 ID:1496 IpLen:20 DgmLen:498 DF
***AP*** Seq: 0xBDD4960A  Ack: 0xAC31A280  Win: 0x776  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5739505 166797609

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:15:06.474654 64.30.224.42:80 -> 192.168.1.169:2634
TCP TTL:63 TOS:0x0 ID:24543 IpLen:20 DgmLen:639 DF
***AP*** Seq: 0x5EA4839  Ack: 0x2CDFA0DE  Win: 0x2180  TcpLen: 20

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:15:08.286168 216.239.113.95:80 -> 192.168.1.169:2650
TCP TTL:63 TOS:0x0 ID:57018 IpLen:20 DgmLen:728 DF
***AP*** Seq: 0x95FC010  Ack: 0x9C6308FA  Win: 0x1A28  TcpLen: 20

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:15:08.361442 138.108.28.10:80 -> 192.168.1.169:2649
TCP TTL:63 TOS:0x0 ID:46682 IpLen:20 DgmLen:545 DF
***AP*** Seq: 0xA06A8C7  Ack: 0xB9C8BDC5  Win: 0x1B28  TcpLen: 20
```

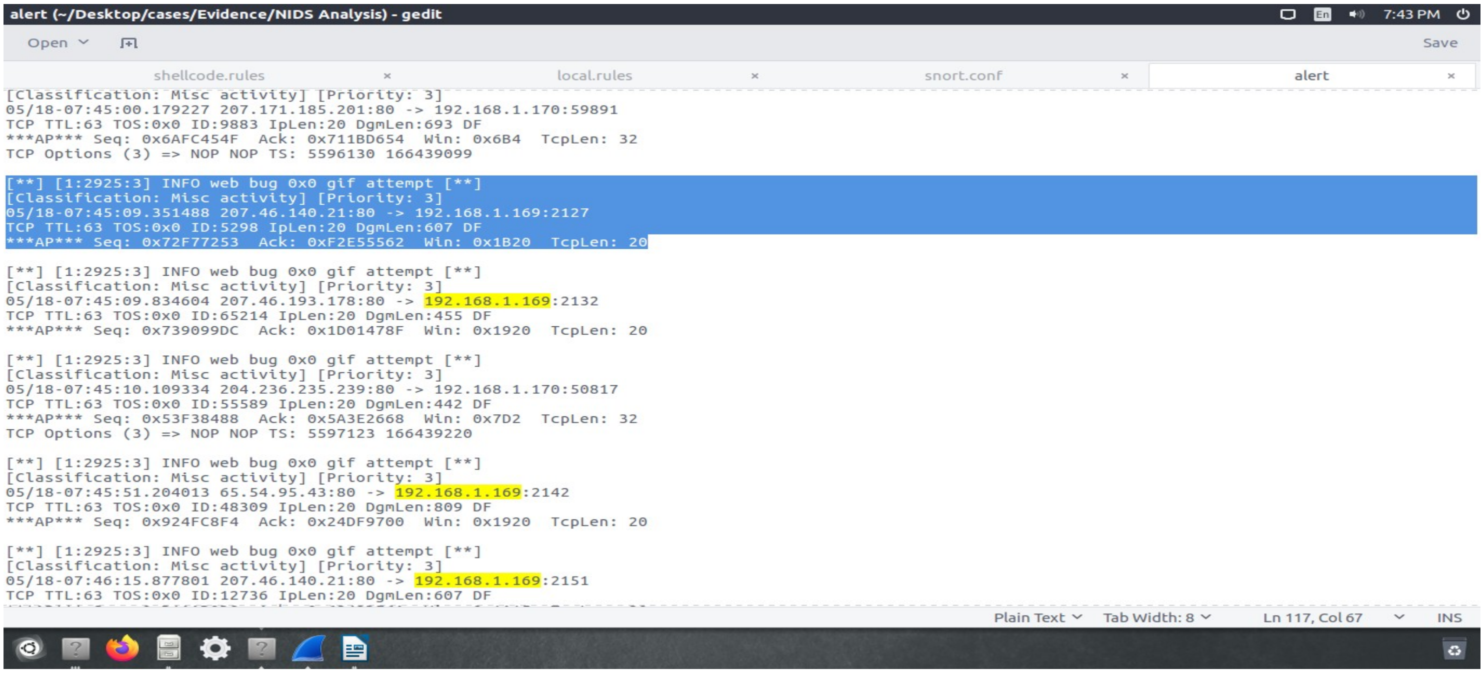
Figure 1.5.4

1.6 Construct a timeline of alerted activities involving the target

Figure 1.6.1, 1.6.2:

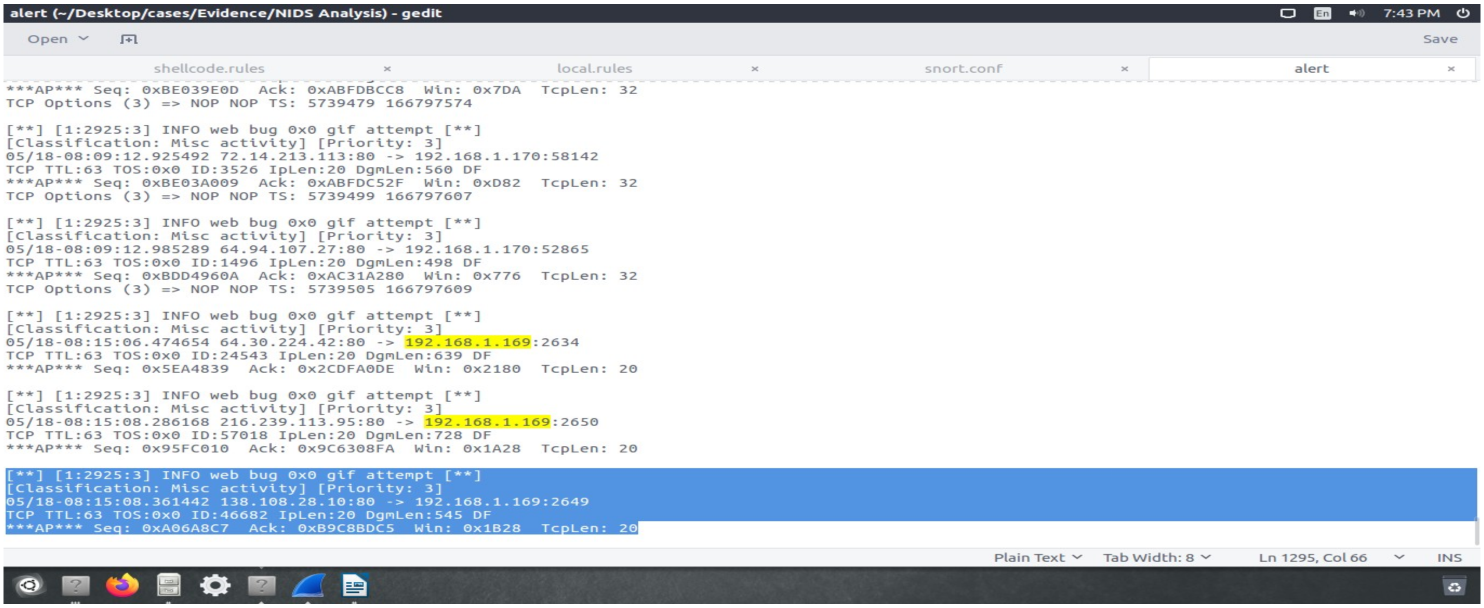
- at 08:01:45 MST we detect "SHELLCODE x86 NOOP [**] [Classification: Executable code was detected]", from an external web server 172.16.16.218:80 to 192.168.1.169. This signifies the malicious intent to inject shellcode.

- and at 08:15:08, we detect "INFO web bug 0x0 gif attempt [**] [Classification: Misc activity]" from different external hosts to internal host 192.168.1.169. This type of alert was also found at many other instances.



```
[Classification: Misc activity] [Priority: 3]
05/18-07:45:00.179227 207.171.185.201:80 -> 192.168.1.170:59891
TCP TTL:63 TOS:0x0 ID:9883 IpLen:20 DgmLen:693 DF
***AP*** Seq: 0x6AFC454F  Ack: 0x711BD654  Win: 0x6B4  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5596130 166439099

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:45:09.351488 207.46.140.21:80 -> 192.168.1.169:2127
TCP TTL:63 TOS:0x0 ID:5298 IpLen:20 DgmLen:607 DF
***AP*** Seq: 0x72F77253  Ack: 0xF2E55562  Win: 0x1B20  TcpLen: 20

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:45:09.834604 207.46.193.178:80 -> 192.168.1.169:2132
TCP TTL:63 TOS:0x0 ID:65214 IpLen:20 DgmLen:455 DF
***AP*** Seq: 0x739099DC  Ack: 0x1D01478F  Win: 0x1920  TcpLen: 20

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:45:10.109334 204.236.235.239:80 -> 192.168.1.170:50817
TCP TTL:63 TOS:0x0 ID:55589 IpLen:20 DgmLen:442 DF
***AP*** Seq: 0x53F38488  Ack: 0x5A3E2668  Win: 0x7D2  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5597123 166439220

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:45:51.204013 65.54.95.43:80 -> 192.168.1.169:2142
TCP TTL:63 TOS:0x0 ID:48309 IpLen:20 DgmLen:809 DF
***AP*** Seq: 0x924FC8F4  Ack: 0x24DF9700  Win: 0x1920  TcpLen: 20

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-07:46:15.877801 207.46.140.21:80 -> 192.168.1.169:2151
TCP TTL:63 TOS:0x0 ID:12736 IpLen:20 DgmLen:607 DF
```

Figure 1.6.1



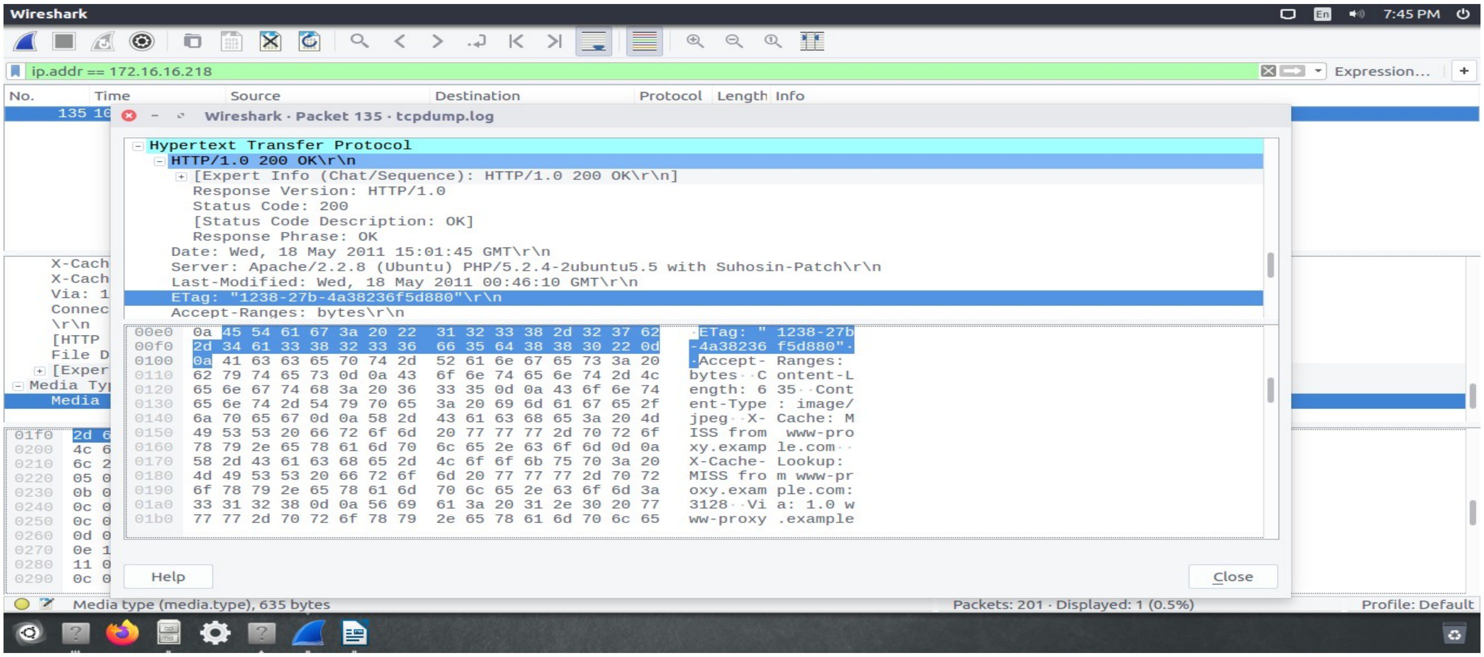
```
***AP*** Seq: 0xBE039E0D  Ack: 0xABFDBCC8  Win: 0x7DA  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5739479 166797574

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:09:12.925492 72.14.213.113:80 -> 192.168.1.170:58142
TCP TTL:63 TOS:0x0 ID:3526 IpLen:20 DgmLen:560 DF
***AP*** Seq: 0xBE03A009  Ack: 0xABFDC52F  Win: 0xD82  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5739499 166797607

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:09:12.985289 64.94.107.27:80 -> 192.168.1.170:52865
TCP TTL:63 TOS:0x0 ID:1496 IpLen:20 DgmLen:498 DF
***AP*** Seq: 0xBDD4960A  Ack: 0xAC31A280  Win: 0x776  TcpLen: 32
TCP Options (3) => NOP NOP TS: 5739505 166797609

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:15:06.474654 64.30.224.42:80 -> 192.168.1.169:2634
TCP TTL:63 TOS:0x0 ID:24543 IpLen:20 DgmLen:639 DF
***AP*** Seq: 0x5EA4839  Ack: 0x2CDFA0DE  Win: 0x2180  TcpLen: 20

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:15:08.286168 216.239.113.95:80 -> 192.168.1.169:2650
TCP TTL:63 TOS:0x0 ID:57018 IpLen:20 DgmLen:728 DF
***AP*** Seq: 0x95FC010  Ack: 0x9C6308FA  Win: 0x1A28  TcpLen: 20

[**] [1:2925:3] INFO web bug 0x0 gif attempt [**]
[Classification: Misc activity] [Priority: 3]
05/18-08:15:08.361442 138.108.28.10:80 -> 192.168.1.169:2649
TCP TTL:63 TOS:0x0 ID:46682 IpLen:20 DgmLen:545 DF
***AP*** Seq: 0xA06A8C7  Ack: 0xB9C8BDC5  Win: 0x1B28  TcpLen: 20
```

Figure 1.6.2

2. Proxy Log Background (Part 2)

2.1 Determine whether the evidence extracted from the Squid cache corroborates our findings from the Snort logs.

Figure 2.1.1, an unusual binary sequence that is commonly associated with buffer overflow exploits was found in ETag in the external webserver's HTTP response : **1238-27b-4a38236f5d880.**



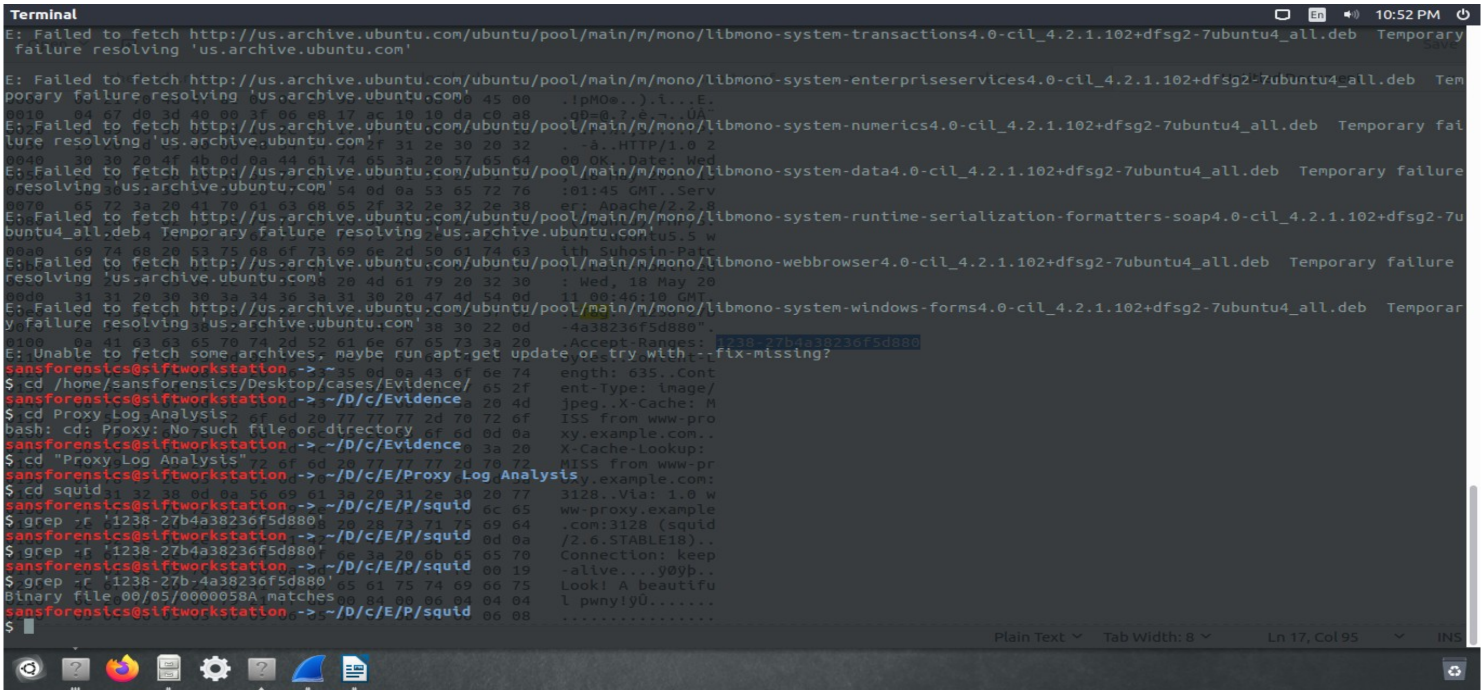
```
Hypertext Transfer Protocol
  HTTP/1.0 200 OK\r\n
    [Expert Info (Chat/Sequence): HTTP/1.0 200 OK\r\n]
    Response Version: HTTP/1.0
    Status Code: 200
    [Status Code Description: OK]
    Response Phrase: OK
  Date: Wed, 18 May 2011 15:01:45 GMT\r\n
  Server: Apache/2.2.8 (Ubuntu) PHP/5.2.4-2ubuntu5.5 with Suhosin-Patch\r\n
  Last-Modified: Wed, 18 May 2011 00:46:10 GMT\r\n
  ETag: "1238-27b-4a38236f5d880"\r\n
  Accept-Ranges: bytes\r\n
```

```
00e0  0a 45 54 61 67 3a 20 22  31 32 33 38 2d 32 37 62   ·ETag: " 1238-27b
00f0  2d 34 61 33 38 32 33 36  66 35 64 38 38 30 22 0d   -4a38236 f5d880"·
0100  0a 41 63 63 65 70 74 2d  52 61 6e 67 65 73 3a 20   ·Accept- Ranges: 
0110  62 79 74 65 73 0d 0a 43  6f 6e 74 65 6e 74 2d 4c   bytes··C ontent-L
0120  65 6e 67 74 68 3a 20 36  33 35 0d 0a 43 6f 6e 74   ength: 6 35··Cont
0130  65 6e 74 2d 54 79 70 65  3a 20 69 6d 61 67 65 2f   ent-Type : image/
0140  6a 70 65 67 0d 0a 58 2d  43 61 63 68 65 3a 20 4d   jpeg··X- Cache: M
0150  49 53 53 20 66 72 6f 6d  20 77 77 77 2d 70 72 6f   ISS from  www-pro
0160  78 79 2e 65 78 61 6d 70  6c 65 2e 63 6f 6d 0d 0a   xy.examp le.com··
0170  58 2d 43 61 63 68 65 2d  4c 6f 6f 6b 75 70 3a 20   X-Cache- Lookup: 
0180  4d 49 53 53 20 66 72 6f  6d 20 77 77 77 2d 70 72   MISS fro m www-pr
0190  6f 78 79 2e 65 78 61 6d  70 6c 65 2e 63 6f 6d 3a   oxy.exam ple.com:
01a0  33 31 32 38 0d 0a 56 69  61 3a 20 31 2e 30 20 77   3128··Vi a: 1.0 w
01b0  77 77 2d 70 72 6f 78 79  2e 65 78 61 6d 70 6c 65   ww-proxy .example
```

```
01f0  2d 6
0200  4c 6
0210  6c 2
0220  05 0
0230  0b 0
0240  0c 0
0250  0c 0
0260  0d 0
0270  0e 1
0280  11 0
0290  0c 0
```

Figure 2.1.1

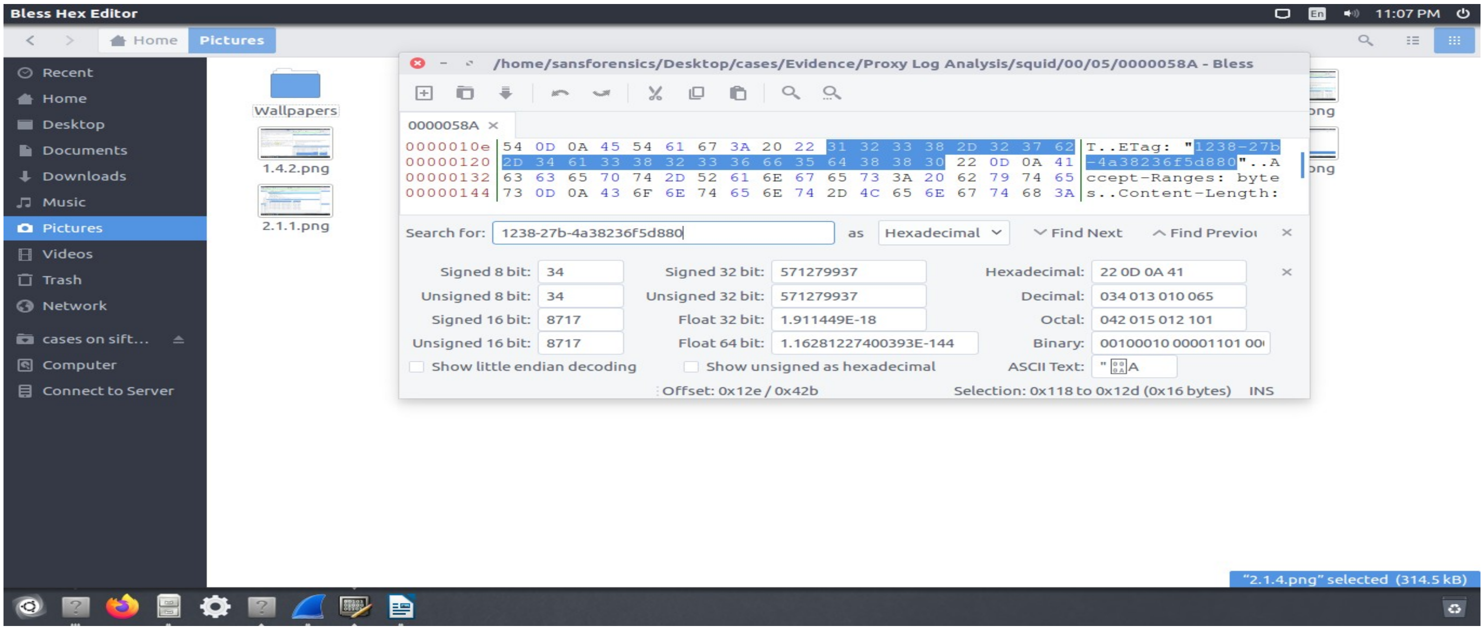
Figure 2.1.2, we use **cd /home/sansforensics/Desktop/cases/Evidence/"Proxy Log Analysis"/squid** and then **grep -r '1238-27b-4a38236f5d880'** to look for file with the matching ETag sequence and we find that a binary file with location "00/05/0000058A" has matched our search.



```
E: Failed to fetch http://us.archive.ubuntu.com/ubuntu/pool/main/m/mono/libmono-system-transactions4.0-cil_4.2.1.102+dfsg2-7ubuntu4_all.deb  Temporary
 failure resolving 'us.archive.ubuntu.com'

E: Failed to fetch http://us.archive.ubuntu.com/ubuntu/pool/main/m/mono/libmono-system-enterpriseservices4.0-cil_4.2.1.102+dfsg2-7ubuntu4_all.deb  Tem
porary failure resolving 'us.archive.ubuntu.com'

E: Failed to fetch http://us.archive.ubuntu.com/ubuntu/pool/main/m/mono/libmono-system-numerics4.0-cil_4.2.1.102+dfsg2-7ubuntu4_all.deb  Temporary fai
lure resolving 'us.archive.ubuntu.com'

E: Failed to fetch http://us.archive.ubuntu.com/ubuntu/pool/main/m/mono/libmono-system-data4.0-cil_4.2.1.102+dfsg2-7ubuntu4_all.deb  Temporary failure
 resolving 'us.archive.ubuntu.com'

E: Failed to fetch http://us.archive.ubuntu.com/ubuntu/pool/main/m/mono/libmono-system-runtime-serialization-formatters-soap4.0-cil_4.2.1.102+dfsg2-7u
buntu4_all.deb  Temporary failure resolving 'us.archive.ubuntu.com'

E: Failed to fetch http://us.archive.ubuntu.com/ubuntu/pool/main/m/mono/libmono-webbrowser4.0-cil_4.2.1.102+dfsg2-7ubuntu4_all.deb  Temporary failure
resolving 'us.archive.ubuntu.com'

E: Failed to fetch http://us.archive.ubuntu.com/ubuntu/pool/main/m/mono/libmono-system-windows-forms4.0-cil_4.2.1.102+dfsg2-7ubuntu4_all.deb  Temporar
y failure resolving 'us.archive.ubuntu.com'

E: Unable to fetch some archives, maybe run apt-get update or try with --fix-missing?
sansforensics@siftworkstation -> ~
$ cd /home/sansforensics/Desktop/cases/Evidence/
sansforensics@siftworkstation -> ~/D/c/Evidence
$ cd Proxy Log Analysis
bash: cd: Proxy: No such file or directory
sansforensics@siftworkstation -> ~/D/c/Evidence
$ cd "Proxy Log Analysis"
sansforensics@siftworkstation -> ~/D/c/E/Proxy Log Analysis
$ cd squid
sansforensics@siftworkstation -> ~/D/c/E/P/squid
$ grep -r '1238-27b4a38236f5d880'
sansforensics@siftworkstation -> ~/D/c/E/P/squid
$ grep -r '1238-27b4a38236f5d880'
sansforensics@siftworkstation -> ~/D/c/E/P/squid
$ grep -r '1238-27b-4a38236f5d880'
Binary file 00/05/0000058A matches
sansforensics@siftworkstation -> ~/D/c/E/P/squid
$
```

Figure 2.1.2

Figure 2.1.3, we open the binary file 0000058A, using Bless Hex Editor and could verify the matching ETag sequence - **1238-27b4a38236f5d880**. We were also able to find the location **http://www.evil.evl/pwny.jpg** of the malicious JPEG file, **pwny.jpg** in ASCII characters.



/home/sansforensics/Desktop/cases/Evidence/Proxy Log Analysis/squid/00/05/0000058A - Bless

```
0000010e 54 0D 0A 45 54 61 67 3A 20 22 31 32 33 38 2D 32 37 62  T..ETag: "1238-27b
00000120 2D 34 61 33 38 32 33 36 66 35 64 38 38 30 22 0D 0A 41  -4a38236f5d880"..A
00000132 63 63 65 70 74 2D 52 61 6E 67 65 73 3A 20 62 79 74 65  ccept-Ranges: byte
00000144 73 0D 0A 43 6F 6E 74 65 6E 74 2D 4C 65 6E 67 74 68 3A  s..Content-Length:
```

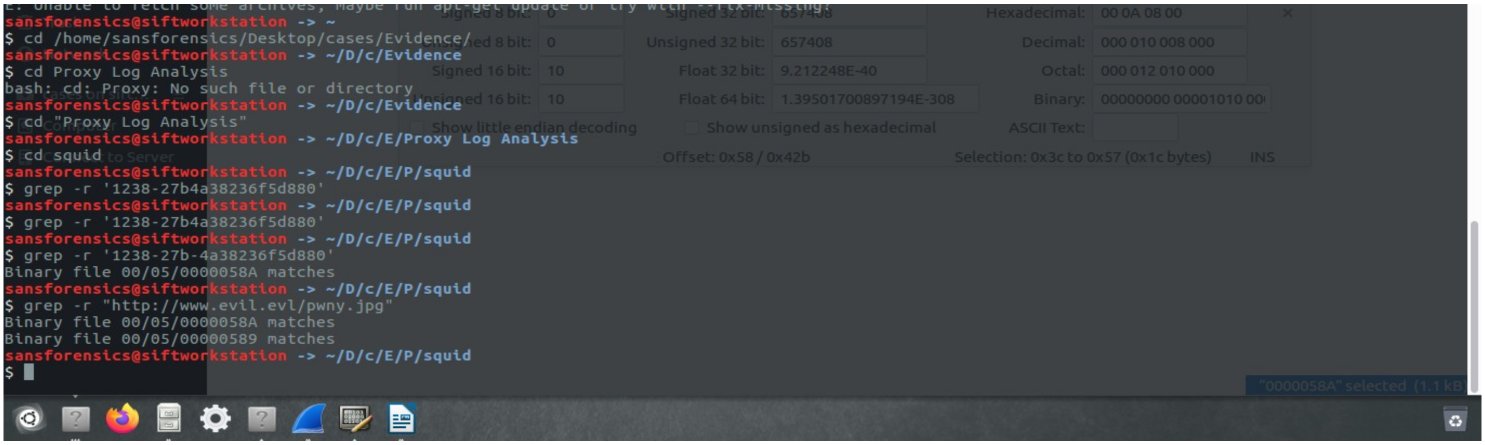
Search for: 1238-27b-4a38236f5d880 as Hexadecimal

Signed 8 bit:	34	Signed 32 bit:	571279937	Hexadecimal:	22 0D 0A 41
Unsigned 8 bit:	34	Unsigned 32 bit:	571279937	Decimal:	034 013 010 065
Signed 16 bit:	8717	Float 32 bit:	1.911449E-18	Octal:	042 015 012 101
Unsigned 16 bit:	8717	Float 64 bit:	1.16281227400393E-144	Binary:	00100010 00001101 00
□ Show little endian decoding		□ Show unsigned as hexadecimal		ASCII Text:	" A

Offset: 0x12e / 0x42b Selection: 0x118 to 0x12d (0x16 bytes) INS

Figure 2.1.3

Figure 2.1.4, using **cd /home/sansforensics/Desktop/cases/Evidence/"Proxy Log Analysis"/squid** and then **grep -r "http://www.evil.evl/pwny.jpg"** we search for the .jpg file's url among other cache files in the squid directory and find two binary file matches **0000058A** and **00000589.**

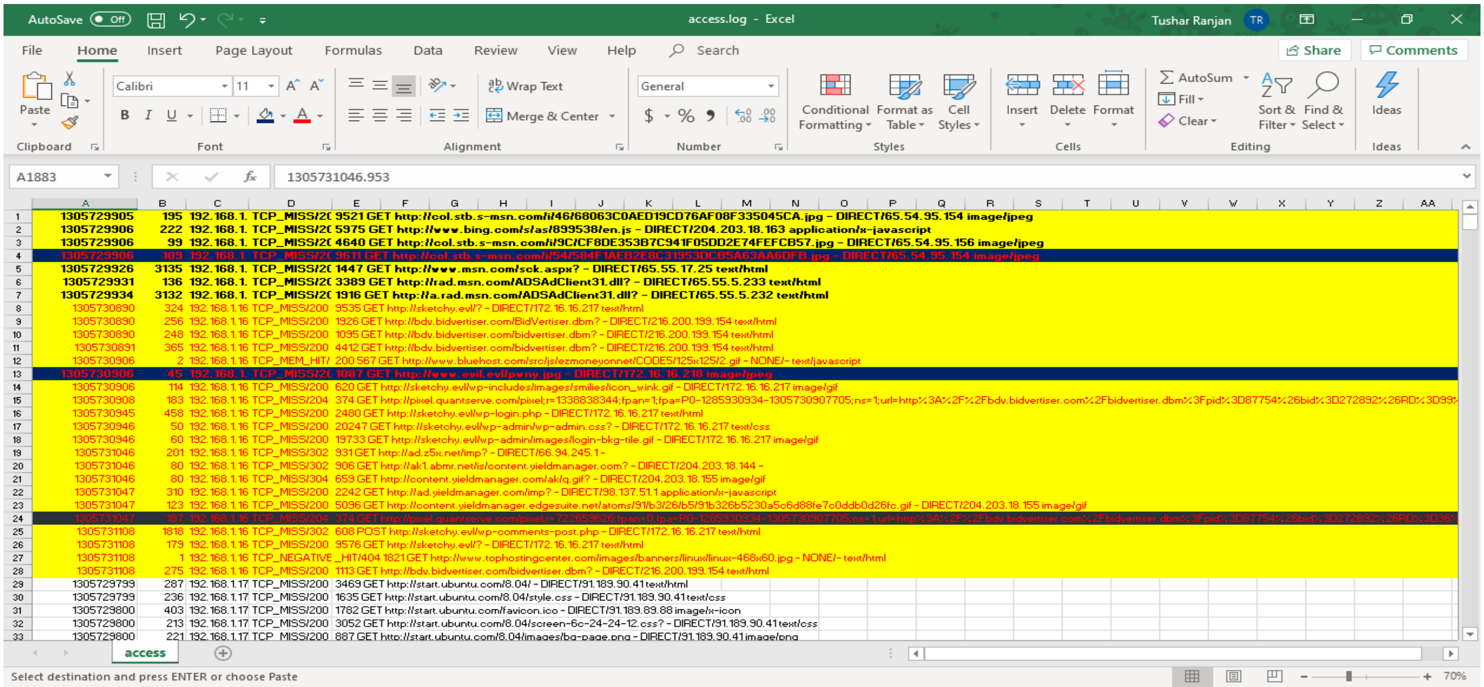


```
E: Unable to fetch some archives, maybe run apt-get update or try with --fix-missing?
sansforensics@siftworkstation -> ~
$ cd /home/sansforensics/Desktop/cases/Evidence/
sansforensics@siftworkstation -> ~/D/c/Evidence
$ cd Proxy Log Analysis
bash: cd: Proxy: No such file or directory
sansforensics@siftworkstation -> ~/D/c/Evidence
$ cd "Proxy Log Analysis"
sansforensics@siftworkstation -> ~/D/c/E/Proxy Log Analysis
$ cd squid
sansforensics@siftworkstation -> ~/D/c/E/P/squid
$ grep -r '1238-27b4a38236f5d880'
sansforensics@siftworkstation -> ~/D/c/E/P/squid
$ grep -r '1238-27b4a38236f5d880'
sansforensics@siftworkstation -> ~/D/c/E/P/squid
$ grep -r '1238-27b-4a38236f5d880'
Binary file 00/05/0000058A matches
sansforensics@siftworkstation -> ~/D/c/E/P/squid
$ grep -r "http://www.evil.evl/pwny.jpg"
Binary file 00/05/0000058A matches
Binary file 00/05/00000589 matches
sansforensics@siftworkstation -> ~/D/c/E/P/squid
$ 
```

Figure 2.1.4

2.2 Present any information you can find regarding the identity of any internal users who have been engaged in suspicious activities.

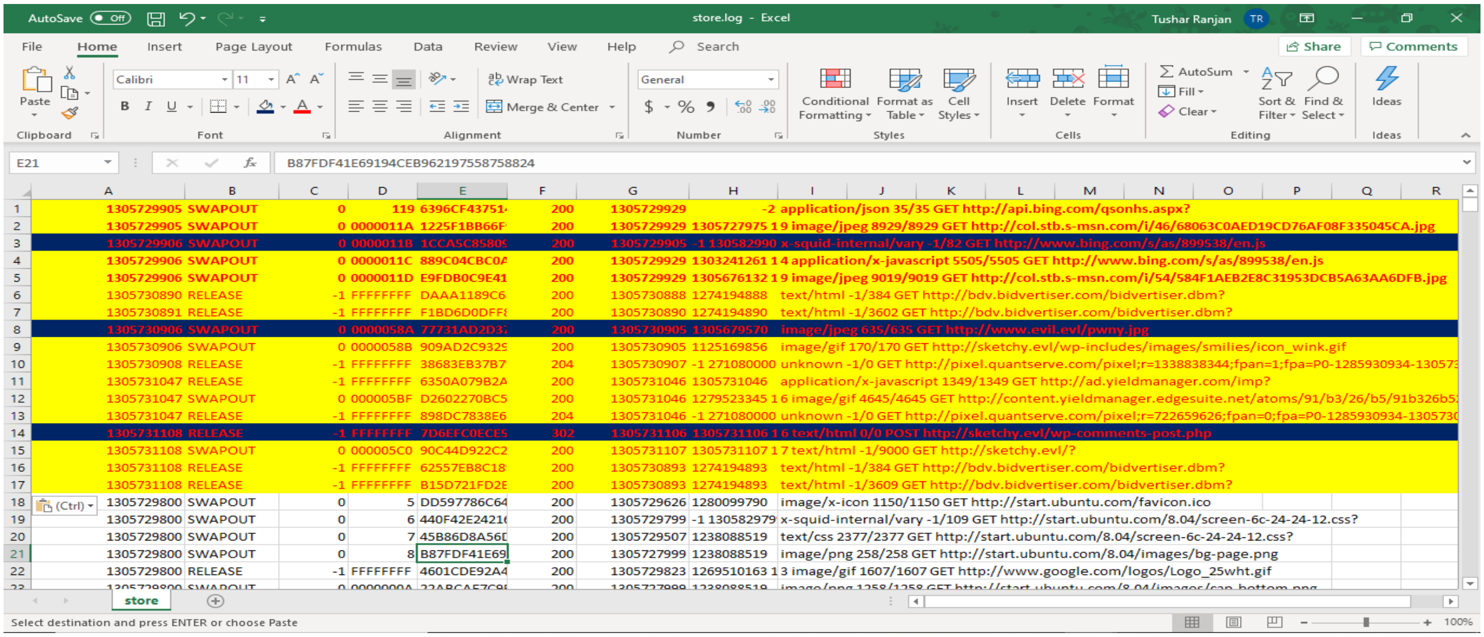
Figure 2.2.1 and 2.2.2, we copy the **access.log** and **store.log** files from **var-log-squid.zip** to a Windows machine to open them using MS Excel and filter them using 192.168.1.169. Then we hit and try different epoch timestamps to match with the alert timestamp. For this we use an online converter (**figure 2.2.3**) and find out that **1305729906**, **1305730906** and **1305731047 (blue color coded)** are the closest timestamps corresponding to **07:45:06** (when internal host 192.168.1.169 started browsing external web sites)**, 08:01:46** (when alert triggered for shellcode injection) and **08:04:07** (when internal host 192.168.1.169 sent crafted packets to other internal hosts).



	A	B	C	D	E
1	1305729905	195	192.168.1.	TCP_MISS/2(	9521 GET http://col.stb.s-msn.com/i/46/68063C0AED19CD76AF08F335045CA.jpg – DIRECT/65.54.95.154 image/jpeg
2	1305729906	222	192.168.1.	TCP_MISS/2(	5975 GET http://www.bing.com/s/as/899538/en.js – DIRECT/204.203.18.163 application/x-javascript
3	1305729906	99	192.168.1.	TCP_MISS/2(	4640 GET http://col.stb.s-msn.com/i/9C/CF8DE353B7C941F05DD2E74FEFCB57.jpg – DIRECT/65.54.95.156 image/jpeg
4	1305729906	109	192.168.1.	TCP_MISS/2(	9611 GET http://col.stb.s-msn.com/i/54/584F1AEB2E8C31953DCB5A63AA6DFB.jpg – DIRECT/65.54.95.154 image/jpeg
5	1305729926	3135	192.168.1.	TCP_MISS/2(	1447 GET http://www.msn.com/sck.aspx? – DIRECT/65.55.17.25 text/html
6	1305729931	136	192.168.1.	TCP_MISS/2(	3389 GET http://rad.msn.com/ADSAdClient31.dll? – DIRECT/65.55.5.233 text/html
7	1305729934	3132	192.168.1.	TCP_MISS/2(	1916 GET http://a.rad.msn.com/ADSAdClient31.dll? – DIRECT/65.55.5.232 text/html
8	1305730890	324	192.168.1.16	TCP_MISS/200	9535 GET http://sketchy.evl/? – DIRECT/172.16.16.217 text/html
9	1305730890	256	192.168.1.16	TCP_MISS/200	1926 GET http://bdv.bidvertiser.com/BidVertiser.dbm? – DIRECT/216.200.199.154 text/html
10	1305730890	248	192.168.1.16	TCP_MISS/200	1095 GET http://bdv.bidvertiser.com/bidvertiser.dbm? – DIRECT/216.200.199.154 text/html
11	1305730891	365	192.168.1.16	TCP_MISS/200	4412 GET http://bdv.bidvertiser.com/bidvertiser.dbm? – DIRECT/216.200.199.154 text/html
12	1305730906	2	192.168.1.16	TCP_MEM_HIT/	200 567 GET http://www.bluehost.com/src/js/ezmoneyonnet/CODE5/125x125/2.gif – NONE/– text/javascript
13	1305730906	45	192.168.1.	TCP_MISS/2(	1087 GET http://www.evil.evl/pwny.jpg – DIRECT/172.16.16.218 image/jpeg
14	1305730906	114	192.168.1.16	TCP_MISS/200	620 GET http://sketchy.evl/wp-includes/images/smilies/icon_wink.gif – DIRECT/172.16.16.217 image/gif
15	1305730908	183	192.168.1.16	TCP_MISS/204	374 GET http://pixel.quantserve.com/pixel;r=1338838344;fpan=1;fpa=P0-1285930934-1305730907705;ns=1;url=http%3A%2F%2Fbdv.bidvertiser.com%2Fbidvertiser.dbm%3Fpid%3D87754%26bid%3D272892%26RD%3D99%
16	1305730945	458	192.168.1.16	TCP_MISS/200	2480 GET http://sketchy.evl/wp-login.php – DIRECT/172.16.16.217 text/html
17	1305730946	50	192.168.1.16	TCP_MISS/200	20247 GET http://sketchy.evl/wp-admin/wp-admin.css? – DIRECT/172.16.16.217 text/css
18	1305730946	60	192.168.1.16	TCP_MISS/200	19733 GET http://sketchy.evl/wp-admin/images/login-bkg-tile.gif – DIRECT/172.16.16.217 image/gif
19	1305731046	201	192.168.1.16	TCP_MISS/302	931 GET http://ad.z5x.net/imp? – DIRECT/66.94.245.1 –
20	1305731046	80	192.168.1.16	TCP_MISS/302	906 GET http://ak1.abmr.net/is/content.yieldmanager.com? – DIRECT/204.203.18.144 –
21	1305731046	80	192.168.1.16	TCP_MISS/304	659 GET http://content.yieldmanager.com/ak/q.gif? – DIRECT/204.203.18.155 image/gif
22	1305731047	310	192.168.1.16	TCP_MISS/200	2242 GET http://ad.yieldmanager.com/imp? – DIRECT/98.137.51.1 application/x-javascript
23	1305731047	123	192.168.1.16	TCP_MISS/200	5096 GET http://content.yieldmanager.edgesuite.net/atoms/91/b3/26/b5/91b326b5230a5c6d88fe7c0ddb0d26fc.gif – DIRECT/204.203.18.155 image/gif
24	1305731047	187	192.168.1.16	TCP_MISS/204	374 GET http://pixel.quantserve.com/pixel;r=722659626;fpan=0;fpa=P0-1285930934-1305730907705;ns=1;url=http%3A%2F%2Fbdv.bidvertiser.com%2Fbidvertiser.dbm%3Fpid%3D87754%26bid%3D272892%26RD%3D36%
25	1305731108	1818	192.168.1.16	TCP_MISS/302	608 POST http://sketchy.evl/wp-comments-post.php – DIRECT/172.16.16.217 text/html
26	1305731108	179	192.168.1.16	TCP_MISS/200	9576 GET http://sketchy.evl/? – DIRECT/172.16.16.217 text/html
27	1305731108	1	192.168.1.16	TCP_NEGATIVE	_HIT/404 1821 GET http://www.tophostingcenter.com/images/banners/linux/linux-468x60.jpg – NONE/– text/html
28	1305731108	275	192.168.1.16	TCP_MISS/200	1113 GET http://bdv.bidvertiser.com/bidvertiser.dbm? – DIRECT/216.200.199.154 text/html
29	1305729799	287	192.168.1.17	TCP_MISS/200	3469 GET http://start.ubuntu.com/8.04/ – DIRECT/91.189.90.41 text/html
30	1305729799	236	192.168.1.17	TCP_MISS/200	1635 GET http://start.ubuntu.com/8.04/style.css – DIRECT/91.189.90.41 text/css
31	1305729800	403	192.168.1.17	TCP_MISS/200	1782 GET http://start.ubuntu.com/favicon.ico – DIRECT/91.189.89.88 image/x-icon
32	1305729800	213	192.168.1.17	TCP_MISS/200	3052 GET http://start.ubuntu.com/8.04/screen-6c-24-24-12.css? – DIRECT/91.189.90.41 text/css
33	1305729800	221	192.168.1.17	TCP_MISS/200	887 GET http://start.ubuntu.com/8.04/images/bg-page.png – DIRECT/91.189.90.41 image/png

Figure 2.2.1

After careful analysis of activities from both access.log and store.log files around the aforementioned timestamps, we find out that the internal user was browsing images on various external websites, among which also happened to show some activity with .evl websites like **http://sketchy.evl/?, www.evil.evl/pwny.jpg** (.evl is a top level domain used by Evil Systems). This user also seems to have admin access to one of the .evl website because there are traces of login attempts to admin profiles.

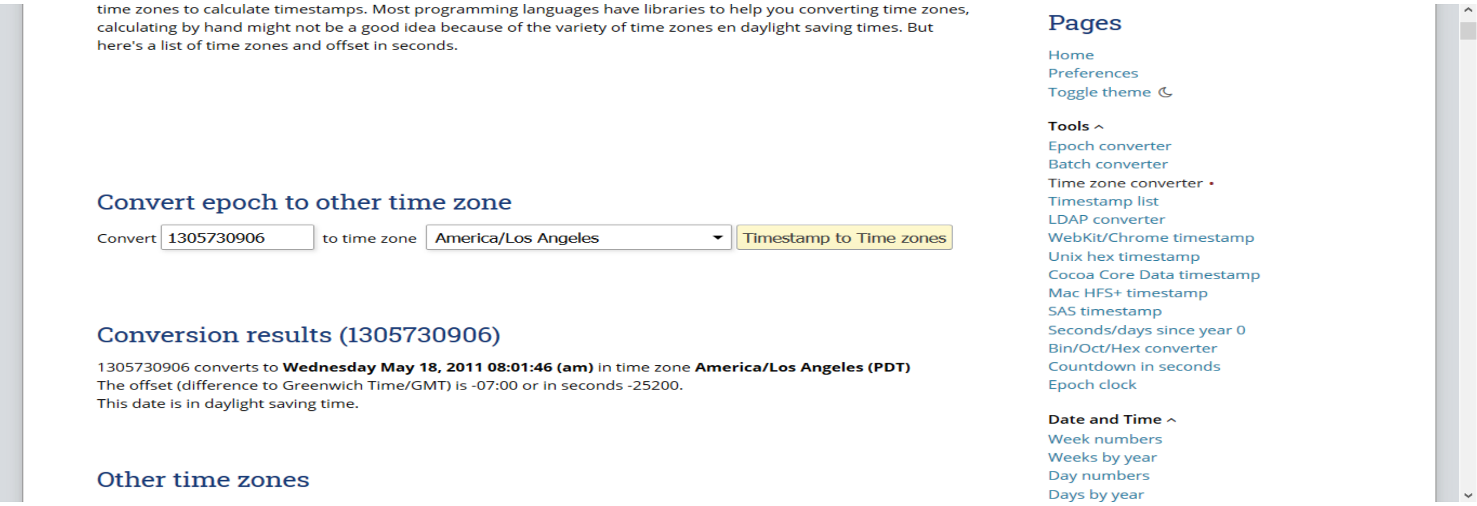


E21 fx B87FDF41E69194CEB962197558758824

	A	B	C	D	E	F	G	H	I
1	1305729905	SWAPOUT	0	119	6396CF43751	200	1305729929	-2	application/json 35/35 GET http://api.bing.com/qsonhs.aspx?
2	1305729905	SWAPOUT	0	0000011A	1225F1BB66F	200	1305729929	1305727975 1	9 image/jpeg 8929/8929 GET http://col.stb.s-msn.com/i/46/68063C0AED19CD76AF08F335045CA.jpg
3	1305729906	SWAPOUT	0	0000011B	1CCA5C8580	200	1305729905	-1 130582990	x-squid-internal/vary -1/82 GET http://www.bing.com/s/as/899538/en.js
4	1305729906	SWAPOUT	0	0000011C	889C04CBC0A	200	1305729929	1303241261 1	4 application/x-javascript 5505/5505 GET http://www.bing.com/s/as/899538/en.js
5	1305729906	SWAPOUT	0	0000011D	E9FDB0C9E41	200	1305729929	1305676132 1	9 image/jpeg 9019/9019 GET http://col.stb.s-msn.com/i/54/584F1AEB2E8C31953DCB5A63AA6DFB.jpg
6	1305730890	RELEASE	-1	FFFFFFFF	DAAA1189C6	200	1305730888	1274194888	text/html -1/384 GET http://bdv.bidvertiser.com/bidvertiser.dbm?
7	1305730891	RELEASE	-1	FFFFFFFF	F1BD6D0DFF	200	1305730890	1274194890	text/html -1/3602 GET http://bdv.bidvertiser.com/bidvertiser.dbm?
8	1305730906	SWAPOUT	0	0000058A	77731AD2D3	200	1305730905	1305679570	image/jpeg 635/635 GET http://www.evil.evl/pwny.jpg
9	1305730906	SWAPOUT	0	0000058B	909AD2C932	200	1305730905	1125169856	image/gif 170/170 GET http://sketchy.evl/wp-includes/images/smilies/icon_wink.gif
10	1305730908	RELEASE	-1	FFFFFFFF	38683EB37B7	204	1305730907	-1 271080000	unknown -1/0 GET http://pixel.quantserve.com/pixel;r=1338838344;fpan=1;fpa=P0-1285930934-13057
11	1305731047	RELEASE	-1	FFFFFFFF	6350A079B2A	200	1305731046	1305731046	application/x-javascript 1349/1349 GET http://ad.yieldmanager.com/imp?
12	1305731047	SWAPOUT	0	000005BF	D2602270BC5	200	1305731046	1279523345 1	6 image/gif 4645/4645 GET http://content.yieldmanager.edgesuite.net/atoms/91/b3/26/b5/91b326b5
13	1305731047	RELEASE	-1	FFFFFFFF	898DC7838E6	204	1305731046	-1 271080000	unknown -1/0 GET http://pixel.quantserve.com/pixel;r=722659626;fpan=0;fpa=P0-1285930934-130573
14	1305731108	RELEASE	-1	FFFFFFFF	7D6EFC0ECE5	302	1305731106	1305731106 1	6 text/html 0/0 POST http://sketchy.evl/wp-comments-post.php
15	1305731108	SWAPOUT	0	000005C0	90C44D922C2	200	1305731107	1305731107 1	7 text/html -1/9000 GET http://sketchy.evl/?
16	1305731108	RELEASE	-1	FFFFFFFF	62557EB8C18	200	1305730893	1274194893	text/html -1/384 GET http://bdv.bidvertiser.com/bidvertiser.dbm?
17	1305731108	RELEASE	-1	FFFFFFFF	B15D721FD2E	200	1305730893	1274194893	text/html -1/3609 GET http://bdv.bidvertiser.com/bidvertiser.dbm?
18	1305729800	SWAPOUT	0	5	DD597786C64	200	1305729626	1280099790	image/x-icon 1150/1150 GET http://start.ubuntu.com/favicon.ico
19	1305729800	SWAPOUT	0	6	440F42E2421	200	1305729799	-1 130582979	x-squid-internal/vary -1/109 GET http://start.ubuntu.com/8.04/screen-6c-24-24-12.css?
20	1305729800	SWAPOUT	0	7	45B86D8A56D	200	1305729507	1238088519	text/css 2377/2377 GET http://start.ubuntu.com/8.04/screen-6c-24-24-12.css?
21	1305729800	SWAPOUT	0	8	B87FDF41E69	200	1305727999	1238088519	image/png 258/258 GET http://start.ubuntu.com/8.04/images/bg-page.png
22	1305729800	RELEASE	-1	FFFFFFFF	4601CDE92A4	200	1305729823	1269510163 1	3 image/gif 1607/1607 GET http://www.google.com/logos/Logo_25wht.gif
23	1305729800	SWAPOUT	0	0000000A	22ABCAE7C9	200	1305727999	1238088519	image/png 1258/1258 GET http://start.ubuntu.com/8.04/images/cap_bottom.png

Select destination and press ENTER or choose Paste

Figure 2.2.2



time zones to calculate timestamps. Most programming languages have libraries to help you converting time zones, calculating by hand might not be a good idea because of the variety of time zones en daylight saving times. But here's a list of time zones and offset in seconds.

Convert epoch to other time zone

Convert 1305730906 to time zone America/Los Angeles [Timestamp to Time zones]

Conversion results (1305730906)

1305730906 converts to **Wednesday May 18, 2011 08:01:46 (am)** in time zone **America/Los Angeles (PDT)**
The offset (difference to Greenwich Time/GMT) is -07:00 or in seconds -25200.
This date is in daylight saving time.

Other time zones

Figure 2.2.3

Conclusion

The NIDS/NIPS and Web Proxy Analysis lab taught me what tools I can use to to to a) Perform log analysis on IDS/IPS alerts (Snort) to become familiar with alert formats and using them to investigate potentially malicious traffic and b) Perform log analysis on Proxy logs (Squid) to become familiar with proxy log formats, and how to review them to investigate web related traffic. This lab also gave me a good idea about how real-world digital evidences are collected and preserved for future use to dig deeper to investigate potential case of an insider threats just by capturing their logs and network activity.

After the analysis I was able to infer that, an insider from the company is using the internal host to browse images on various external sources, with one of them being a highly malicious domain. There were also the traces of admin level logins to .evl domain leading to a malicious .jpg file trying to inject shellcode.

Glossary

1. Network Intrusion Protection Systems (NIPS) and Network Intrusion Detection Systems (NIDS) are tested on the Technologies and Tools portion of the Security+ certification exam. This article details what is covered on the Security+ certification exam regarding these important network security devices. This article should not substitute for studying but rather serve as a brief review and guide for areas that you may need to look over again.
2. Firewall Analyzer (Proxy Log Analyzer) collects and archives the proxy server logs, analyzes them, and generates useful corporate internet access information reports. Proxy server reports provide network security administrators and managed security service providers (MSSP) with important insight into the efficiency of their corporate Internet usage. As a proxy log analysis tool, Firewall Analyzer supports BlueCoat, Microsoft ISA, Squid proxy logs and servers.
3. Wireshark is the world's foremost and widely used network protocol analyzer. It lets you see what's happening on your network at a microscopic level and is the de facto (and often de jure) standard across many commercial and non-profit enterprises, government agencies, and educational institutions. Wireshark development thrives thanks to the volunteer contributions of networking experts around the globe and is the continuation of a project started by Gerald Combs in 1998.
4. grep is a command-line utility for searching plain-text data sets for lines that match a regular expression. Its name comes from the ed command g/re/p, which has the same effect: doing a global search with the regular expression and printing all matching lines.
5. Snort is an open source network intrusion detection system (NIDS) created by Martin Roesch. ... Through protocol analysis and content searching and matching, Snort detects attack methods, including denial of service, buffer overflow, CGI attacks, stealth port scans, and SMB probes.

References

[1] “Technologies And Tools - NIPS / NIDS.” Infosec Resources, resources.infosecinstitute.com/category/certifications-training/securityplus/sec-domains/technologies-and-tools-in-security/installing-and-configuring-network-components-to-support-organizational-security/technologies-and-tools-nips-nids/.

[2] Hoffman, Chris. “How to Use Wireshark to Capture, Filter and Inspect Packets.” How, How-To Geek, 14 June 2017, www.howtogeek.com/104278/how-to-use-wireshark-to-capture-filter-and-inspect-packets/.

[3] “Grep Command in Unix/Linux.” GeeksforGeeks, ., 20 May 2019, www.geeksforgeeks.org/grep-command-in-unixlinux/.

[4] Rouse, Margaret. “What Is Snort? - Definition from WhatIs.com.” SearchMidmarketSecurity, TechTarget, 21 Sept. 2005, searchmidmarketsecurity.techtarget.com/definition/Snort.